AZ ÉV **GÉPEI** •
LEMEZ**MEGHAJTÓK** •
GEOS **LEÍRÁS** •
LEMEZDOBOZ **FUSIBAN** •

C újság

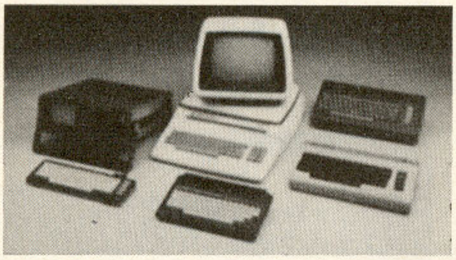
1987/1

AZ
ORSZÁGOS COMMODORE EGYESÜLET
TAGJAINAK



Fejlesztés

A TARTALOMBÓL



Az Országos Commodore Egyesület módszertani kiadványa
Index: ISSN 0237-756 X
Felelős kiadó: Horváth Judit, az egyesület elnöke
Felelős szerkesztő: Pogány György
Szerkesztő: Angyalosi László
Művészeti szerkesztő: Várhelyi László
Fotó: Gál Imre
Szerkesztőségi titkár: Tóth Éva
Készült: a Globál GMK gondozásában
Levélcím: Commodore újság
Pozsonyi út 50. fsz. 4.
1133
Telefon: 408-603
ELEKTRO-COOP Nyomda -86319
Felelős vezető: Szathmáry Miklós

1987-es TAGDÍJBEFIZETÉS

Ahogyan már az Elnökség határozatában olvashatták, az 1987-es év tagsági díjait kétféleképpen lehet befizetni. Félévre érvényes tagsági igazolványokat nyomattunk, így mód van arra, hogy csak az első félévet fizesse valaki, és arra is, hogy az egész évet egyben fizesse be (ezek a tagjaink előre megkapják a második félév igazolványát is.)

AZ EGYESÜLET ÁLTAL NYÚJTOTT LEHETŐSÉGEK, SZOLGÁLTATÁSOK VÁLTOZATLANOK!

DEÁKPÁHOLY

negyedévenként egy kivonat a Commodore újságból
- bizonyos kedvezmények, szolgáltatások
- pötyögő szolgálat
- ingyen apróhirdetési lehetőség a C-újságban

TAGSÁGI DÍJ:

egész évre	**116,– Ft.**
fél évre	**60,– Ft.**

PLUSZPÁHOLY

valamennyi az előzőekben felsorolt szolgáltatás és kedvezmény
- a havonta megjelenő 36 oldalas Commodore újság
- egyéb vásárlási kedvezmények

TAGSÁGI DÍJ:

egész évre	**1264,– Ft.**
fél évre	**650,– Ft.**

SZUPERPÁHOLY

valamennyi az előzőekben felsorolt szolgáltatás és kedvezmény
- havonta 14 példány a Commodore újságból
- egyéb vásárlási kedvezmények

TAGSÁGI DÍJ:

egész évre	**12 128,– Ft.**
fél évre	**6 100,– Ft.**

A tagsági díjak befizetési módja sem változott, a legegyszerűbb megoldás rózsaszínű postai utalványon az Egyesület számlájára, illetve közületek részére banki átutalással (kívánságra számlát küldünk). Akik az év folyamán fizetik be a tagdíjat, visszamenőleg megkapják az újságot, és igénybe vehetik azokat a szolgáltatásokat, kedvezményeket, amelyek még nem jártak le. Ha valamelyik egyesületi tagunk kíván magasabb páholyba lépni, úgy csak a különbözetet kell befizetnie számlánkra.
Akik 1987. február végéig befizették a tagsági díjat, azok tagsága folyamatos, sorszámukat megtartják. Akik később fizetik a tagdíjat, új tagsági sorszámot kapnak.
Egyesületünk OTP száma nem változott, kérjük, ezt pontosan tüntessék fel a postai utalványokon. Pontosan és olvashatóan töltsék ki a feladó adatait, hiszen ez alapján postázzuk az újságot. Sajnos ezzel az előző félévben rengeteg gondunk volt, néhány tagunkat máig sem tudtuk azonosítani, mert pontatlan címet adtak meg, vagy – ilyen is volt – egyáltalán nem írtak címet! Az OTP szám a következő:

OTP XIII., VISEGRÁDI U. 7/b.
MNB 217–98292
OTP 565–3610

A személyi- és mikroszámítógépek még nem is olyan hosszú múltra visszatekintő történetében már hagyománynak számít az év végi szavazás, amelyben a témával foglalkozó újságírók, szerkesztőségek – végül is a piactól függetlenül véleményt mondanak. Az „Év számítógépe" tehát nem egyfajta slágerlista, amely a vásárlók, a piac véleményét tükrözi, nem a gépek eladási eredményei alapján áll össze, hanem a konstrukciók műszaki és szellemi színvonalát értékelő szakemberek szavazataiból alakul ki a végső sorrend. A szavazás során az újságírók szinte minden géptípust figyelembe vettek, amely minimális számban már piacra került, és amelynek a rendszerét nem kezelték annyira titkosan, hogy ne lehessen véleményt alkotni róla.

SZEMÉLYI SZÁMÍTÓGÉPEK

IBM PC/AT

Műszaki adatok

Processzor:	Intel 80286
Órajel frekvenciája:	6 vagy 8 MHz
Memóriaméret:	512 vagy 640 KByte RAM 64 KByte ROM
Háttértárak:	1 db 1,2 MByte-os floppy meghajtó 1 db 20 vagy 40 MByte-os keménylemezes tár
Operációs rendszer:	DOS 3.1 verzió vagy Xenix

Helyezések, pontszámok

IBM PC/AT	190 pont
Schneider Joyce	70 pont
Olivetti M28	60 pont
HP Vectra	50 pont
NCR P8	30 pont
Wang APC	20 pont
Siemens PC–D	20 pont
Apricot XN	20 pont
Apple Macintosh +	5 pont
Compaq Deskpro 286	5 pont



HÁZI SZÁMÍTÓGÉPEK

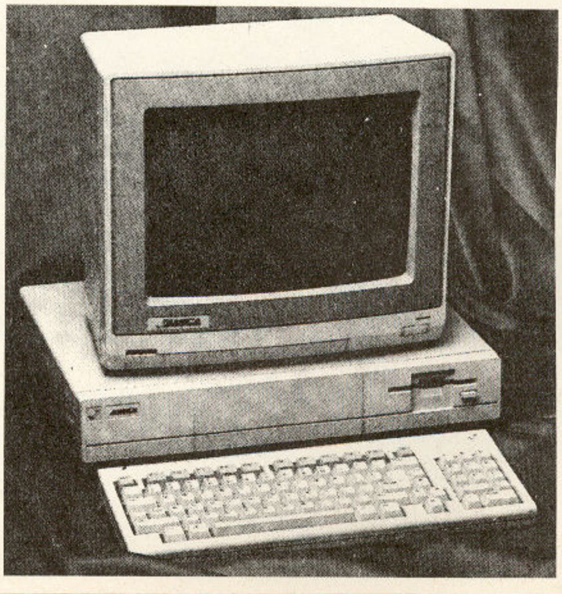
COMMODORE AMIGA

Műszaki adatok

Processzor:	Motorola 68000
Órajel frekvenciája:	7,16 MHz
Memóriaméret:	256 vagy 512 KByte RAM 192 KByte ROM
Háttértárak:	1 db 3,5 collos floppy meghajtó 880 KByte kapacitással (egy külső meghajtó külön kapható)
Grafikai lehetőségek:	640 × 400 képpont 4096 színből egyszerre 16 használható Sprite technika
Hang:	4 csatornán programozható, sztereó jelkimenet, beszédszintetizátor
Operációs rendszer:	Commodore Amiga

Helyezések, pontszámok

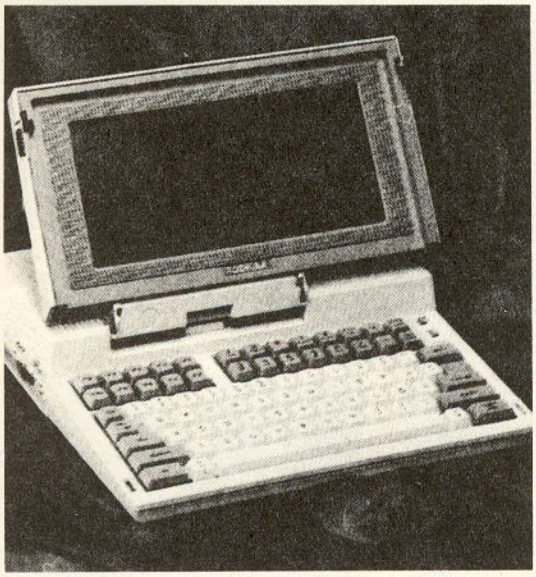
Commodore Amiga	230 pont
Atari 1040 ST	140 pont
Schneider CPC 6128	60 pont
Apple II	60 pont
Commodore 128	40 pont
Commodore 64	40 pont



AZ ÉV SZÁMÍTÓGÉPEI 1986

A szavazásban résztvevő szerkesztőségek:

Personal Computing (USA)
Practical Computing (Nagy Britannia)
CHIP (Olaszország)
Micros (Spanyolország)
Svet Komputerja (Jugoszlávia)
CHIP/Micro Mix (Hollandia)
CHIP (NSZK)



HORDOZHATÓ SZÁMÍTÓGÉPEK

TOSHIBA T3100

Műszaki adatok

Processzor:	Intel 80286
Órajel frekvenciája:	8 MHz
Memóriaméret:	640 KByte
Háttértárak:	1 db 3,5 collos floppy meghajtó, 720 KByte kapacitással 10 MByte-os keménylemezes tár külön vásárolható
Képernyő:	Egyszínű (piros-fekete) plazmaképernyő, 640 × 200 képpont
Operációs rendszer:	Toshiba T3100
Súly:	6,9 kilogramm

Helyezések, pontszámok

Toshiba T3100	300 pont
Compaq portable II	210 pont
Sharp PC 7000	90 pont
Toshiba T2100	70 pont
Data General 1/Mod.2	70 pont
Olivetti M21	30 pont
Panasonic JB–3300	30 pont



KÉZI SZÁMÍTÓGÉPEK

TOSHIBA T1100

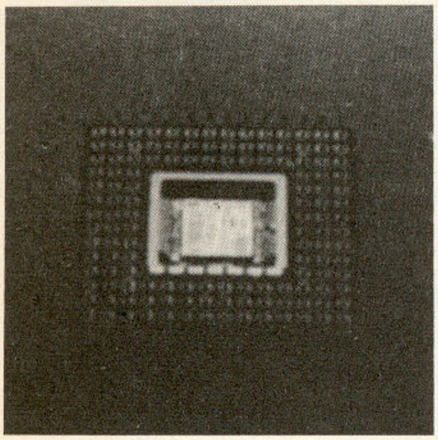
Műszaki adatok

Processzor:	Intel 80C88
Memóriaméret:	256 KByte
Háttértár:	1 db 3,5 collos floppy meghajtó 720 KByte kapacitással
Képernyő:	Folyékony kristályos kijelzés, 80 × 25 karakter 640 × 200 képpont
Operációs rendszer:	Toshiba T1100
Súly:	4,1 kilogramm

Helyezések, pontszámok

Toshiba T1100	180 pont
Zenith Z-171	150 pont
IBM Convertible	100 pont
Kaypro 2000	70 pont

CSÚCSTECHNOLÓGIA



A világpiacon megjelent a szuperchip, amely egy megabit (1 048 576 bit) mennyiségű információ tárolására alkalmas. A digitális információtechnikában használatos kettes számrendszerből adódóan egy bit 0 és 1 állapotot vehet fel. Az egy megabites chip így jó egymilliós tárolási állapottal rendelkezik, azaz több, mint 130 ezer karakternyi szöveget tárolhat. Lehetővé teszi egy kisebb könyv tartalmának rögzítését egy 10,5 × 7,7 mm-es méretű szilíciumlapkán, amely 13 025 bit/mm² tárolási sűrűségnek felel meg.

A 80,85 mm²-es memóriachip négy blokkos szervezésű, és a blokkok egyenként 256 Kbyte-osak. Az adatok közvetlen elérésűek és az üzemeltetéshez csupán 5 voltos feszültség is elegendő. Teljesítményfelvétel működés közben 0,5 W és nyugalmi állapotban pedig 50 mW. Az elérési idő 75 Celsius foknál max. 150 nsec és a ciklusidő körülbelül 285 nsec.

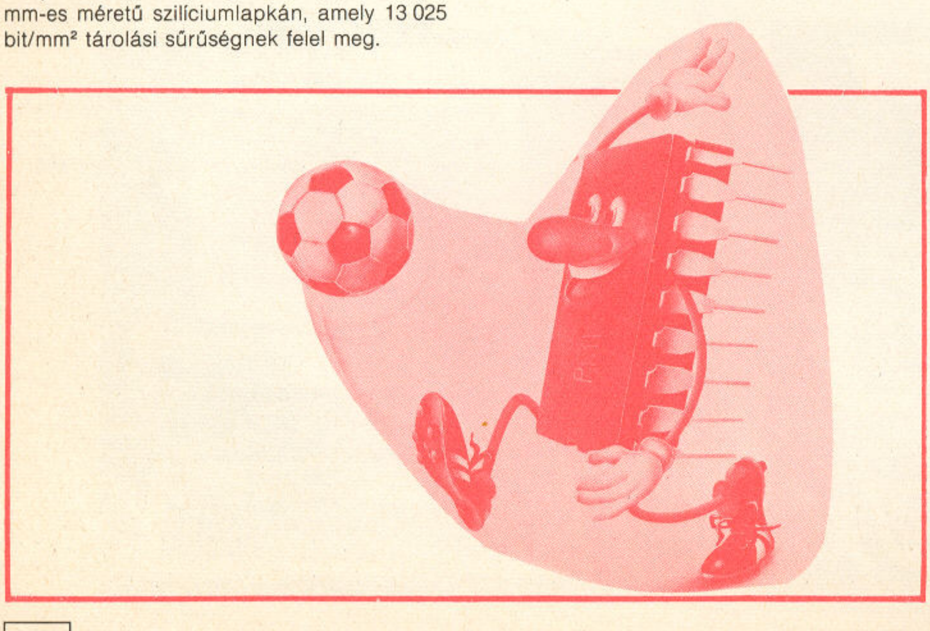
A szilíciumlapkán egy memóriacella, ill. egy bit 45 mikrométeres területet foglal el, a 64 Kbites chipnél ez 170 mikrométert igényel – közel a négyszeresét. A tárolócella fizikai méretének csökkentése függ az IC-t alkotó rétegek vastagságától, ami gyártási technológia kérdése. Az egy megabites szuperchip a jelen csúcstechnológiájának eredménye. A chip 15 nm-es vastagságú (vékonyságú?) rétegek egymásra integrálásával készül. A 64 Kbites chipnél a rétegek vastagsága 25 nm.

A technológiai műveletek során a legkisebb porszem is már rövidzárat és téves előhívást okozhat. Különösen a 0,1 mikronos porszemcsék a kritikusak.

A chipek előállítása szupertiszta helyiségekben történik. Ezek a munkahelyek nyugatnémet szabvány szerint a 10-es tisztasági osztályba tartoznak. A 10-es besorolás azt jelenti, hogy 28 liter levegőben nem lehet több, mint 10 részecske, amelyek egyenként max. 0,5 mikrométeresek. Nem ritka az a munkahely sem, amely a 100-as osztályba tartozik. A gyártási folyamatot jórészt robotok végzik, amelyek tevékenységét monitorokon keresztül ellenőrzik. A kijelzőkön követhetők a részecskék útjai, amelyek lehetővé teszik az esetleges korrigáló intézkedések kezdeményezését. A részecskék ellenőrzése műszeresen is történik, ún. részecskeszámlálóval. A zavaró részecskék arról ismerhetők fel, hogy a lézersugárban reflektálódnak. A kisebb részecskék kimérésére további elektronikus készülékeket használnak.

A rétegek kimunkálását követően a szilícium korongot szétvagdalják. Az egyes chipeket 30 percig tesztelik, amely során mintegy 100 000 órás működést szimulálnak. A próba után kerül csak sor további technológiai lépésre.

Egy MByte-os memóriakapacitás hozható létre 8 egy megabites chip párhuzamos kapcsolásával és már egy 10 MByte-os keménylemezes tárt is kiválthatunk 80 chippel. Az ilyen memóriaegységet tartalmazó számítógépek elérési ideje 150 nsec lesz, összehasonlítás végett egy gyors keménylemezes tár közepes elérési ideje 28 msec. Az integrált áramkörös tár tehát majdnem 190 000-szer gyorsabban működik.



Az egy megabites chip műszaki jellemzői

Memóriakapacitás	1 048 576 bit
Szervezése	256 Kbit*4
	512 Kbit*2
	1 Mbit*1
Memóriatípus	dinamikus RAM
Tápfeszültség	5 V
Teljesítményfelvétel	
üzemi	0,5 W
nyugalmi	50 mW
Elérési idő	150 ns
Chip-méret	10,5*7,7 mm
Keramikus tokozás, 22 kivezetés	

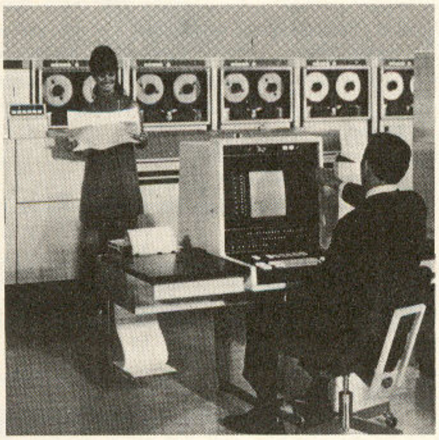
GENERÁCIÓK

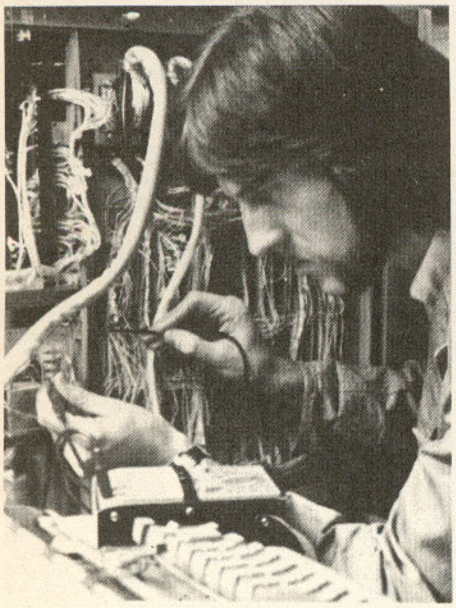
Mottó: Egy újszülöttnek minden gép új.

A számítástechnika első lépéseinek megtétele óta bizony néhány generáción már túljutottunk. A mai Commodore tulajdonosok nagy része tizenéves, tehát a számítógép-használók új generációja is felnőtt, de a gépek is változtak valamicskét.

Amit a képen látnak, a maga korában éppoly korszerű volt, mint a mai mikrogépek. Ráadásul tudásában is közelállt mondjuk egy Commodore 64-eshez. A prospektus, amelyben a fénykép szerepel, éppen tízéves. Az NCR GmbH által gyártott Century 8200 típusú számítógépet hirdeti, irodai felhasználásra.

A gép alapkiépítésben 32 K-s, ebből 16 K a RAM. Központi egysége abban a konyhaszekrény méretű dobozban kapott helyett, amely a fotón a monitortól jobbra látható. Az ismertető szerint a memória bővíthető 48, illetve 64 K-ra is – nyilván újabb szekrények illesztésével.



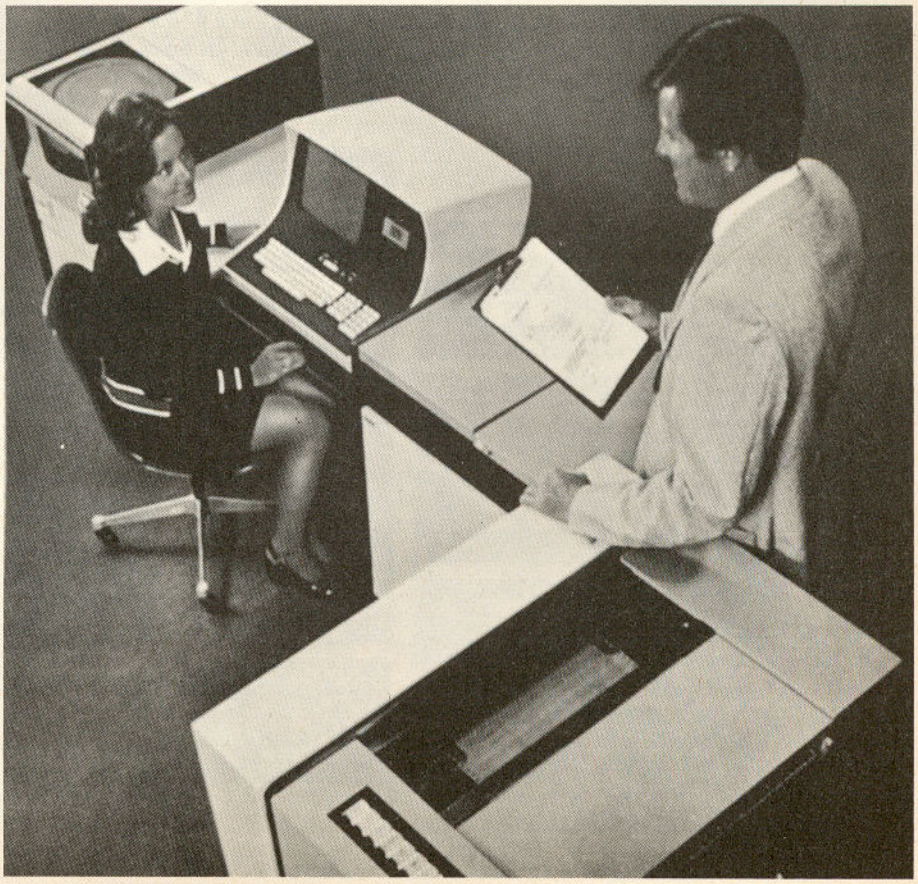


A képernyőn 64-féle karakter jeleníthető meg, a 24 sor mindegyikében 80. Az adatok rögzítésére a beépített kazettás egység szolgál, de csatlakoztatható a géphez lemezegység is – ez a kép bal felső sarkában látható –, valamint nyomtató, amire az irodavezető támaszkodik.

A prospektus megjelenése óta eltelt tíz év alatt sok minden megváltozott. Ma hasonló konfiguráció bemutatásához jóval kisebb távolságra kell beállítani a fényképezőgépet – hiszen egy-egy irodai gép a szükséges perifériákkal együtt elfér egyetlen íróasztalon is. A kiépített rendszer térfogata alig egy százaléka a tíz évvel ezelőttinek! Ráadásul még a C 64, amely nem kifejezetten irodai gép – is többet tud a fotón látható számítógépnél.

Mi lehet a fejlődés további iránya? A méretek csökkentésének ebben a kategóriában nincs értelme, ennek határt szab az, hogy a billentyűzetnek jól kezelhetőnek, a képernyőnek jól láthatónak, a nyomtatott papírnak pedig adott nagyságúnak kell lennie, így inkább az várható, hogy a számítógépek nyelve változik majd, még emberközelibbé, még könnyebben kezelhetővé válnak.

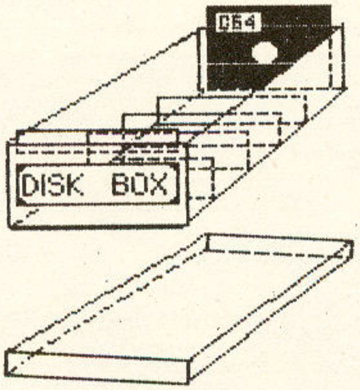
Remélhetőleg hamarosan már azokat a gépeket használjuk, amelyek programnyelv nélkül, akár élőszóval is vezérelhetőek. S ha így lesz, jót nevetünk majd úgy tíz év múlva a Commodore billentyűzetén, esetleg még képernyőjén is. No nem beszélve a lustaságáról, észbeli korlátairól!



Lemez tároló doboz

A lemezegységgel dolgozók gyakorlatból tudják, hogy az adathordozó mágneslemez mennyire sérülékeny. Óvni kell a mágneses tértől, mindenféle szennyezéstől, a hőtől, de még a nap sugárzásától is. A munkaasztalon szerteszét heverő lemezek könnyen áldozatul eshetnek egyetlen vigyázatlan mozdulatunknak is, pedig egyszerű módon megvédhetjük őket.

Erre a célra szolgálnak a nálunk is kapható különféle méretű lemeztároló dobozok, amelyek azonban árukhoz képest egy kicsit keveset nyújtanak: mindössze lemezeket lehet bennük tárolni. Ha rendelkezünk egy kis kézügyességgel és türelemmel, akkor a mellékelt rajz alapján bárki olcsón elkészítheti otthon a saját dobozát.



NÉHÁNY TANÁCS

1. A doboz alapanyagául a barkácsboltokban kapható és könnyen megmunkálható polystirolt (plexit) válasszuk!

2. A szükséges szerszámok: **ceruza, vonalzó, egy pontos derékszög és egy éles kés!** Az elemek kiszabásánál használjunk derékszöget, sok bosszúságot előzünk meg vele.

3. A méreteket rajzoljuk a műanyag lapra, majd éles késsel vágjuk ki, vagy a bejelölés után (ha a vágás helyét kellően kikarcoltuk) az asztal szélén letörhetjük a lemezeket úgy, ahogyan az üvegvágásnál szokták.

Aki nem bízik az ajánlott módszerben, az a műanyaglap feldarabolását egyszerű vasfűrész lappal is elvégezheti.

4. Az összeállítást az alaplap és az egyik oldallap összeragasztásával kezdjük. Második lépésként az elhatároló lapocskákat ragasszuk a helyükre. Ezután következik a másik oldallap és utoljára a két véglap. A derékszöget minden esetben gondosan mérjük ki! Természetesen a rajztól eltérő, tetszés szerinti kiosztást is alkalmazhatunk.

5. A ragasztáshoz használjunk univerzális pillanatragasztót. Ebből többféle is forgalomban van. Közülük jó minőségű és nem túl drága a magyar gyártmányú egykomponenses FERROBOND. (A száradás után mindkét ragasztott felületet érdemes még egyszer vékonyan átkenni!)

6. A kész dobozt a ragasztást követően 1 óra múlva már használhatjuk.

Ha nem találjuk dobozunkat elég erősnek, akkor 8 db, 2 cm széles, derékszögben meghajlított alumínium csíkkal erősítsük meg a dobozt az éleinél.

Az így elkészült doboz 50–100 lemez kényelmes és biztonságos tárolására szolgál.

7. A doboztető összeállítása a dobozéhoz hasonlóan történik. A tető elmozdulása ellen úgy védekezhetünk, hogy a kész doboz két végére (a D elemekre) a G elemet belülről felragasztjuk.

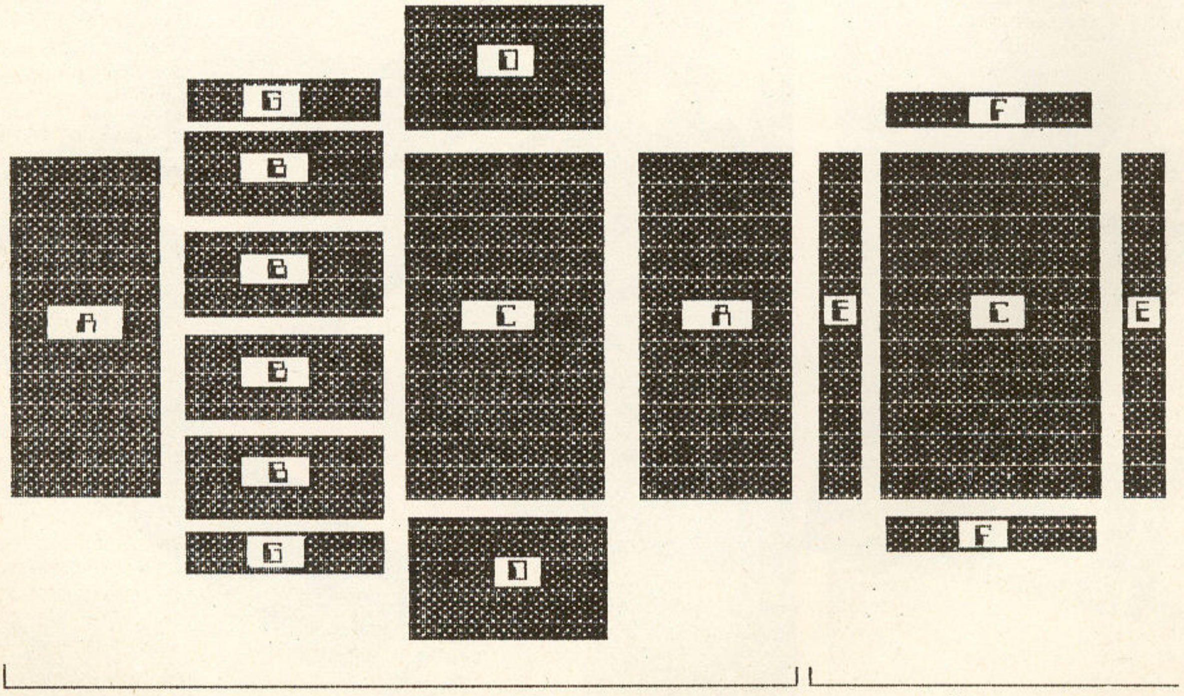
8. Ha a dobozt le akarjuk festeni, használjuk a MATTSCHWARZ-ot, de a NEOLUX is jól megfelel.

Ha felirattal is el akarjuk látni, akkor a papírboltokban kapható ALPHASET betűkészlet használatát ajánlom. A feliratozás után a dobozt érdemes átfújni színtelen AKRILAN lakkal.

HONTI TAMÁS

Elem méretek:

A = 110 x 302 x 4 mm
B = 147 x 65 x 2 mm
C = 155 x 302 x 4 mm
D = 155 x 114 x 4 mm
E = 30 x 302 x 4 mm
F = 155 x 34 x 4 mm
G = 151 x 34 x 4 mm



Doboz elemek: **Tető elemek:**

NYÚZÓPRÓBA



LEMEZTESZT

Sajnos – szokás szerint – kissé késve érkezett meg az újság, így csak most tudok reagálni a „Nyúzópróba felhívásra".

Valamivel több mint 3 éve dolgozom C 64 mikrogéppel és VC 1541-es lemezegységgel. Ez idő alatt több-kevesebb floppy lemezt is (el)használtam. Összesen 4 típussal dolgoztam/dolgozom, ezek a következők:

1. BASF SS–SD
2. WABASH SS–DD
3. SENTINEL SS–SD, SS–DD, DS–DD
4. PARROT SS–DD

Ezekről szeretném most saját – alkalmasint mások által is megerősített – tapasztalataimat leírni. Számomra a legtöbb problémát a BASF lemezek okozták.

Munkám során egy program/programcsomag fejlesztésekor 2 munkalemezt használok párhuzamosan, 1–1 programnak általában 2–3 régebbi változatát is megőrzöm. Lemezeimnek – azok minőségétől függetlenül – mindig csak egyik oldalát használom. BASF lemezek esetén többször is előfordult (néhány hónap alatt 15–20 esetben is – alig használt lemezzel is), hogy pl. ha a „PRG–2" nevű programot akartam a lemezen felülírni, akkor a „PRG–2" program helyett a „PRG" vagy a „PRG–1" program íródott felül, a „PRG–2" program előző változata pedig vagy megmaradt, vagy nem. **Próbáltam a hiba okát keresni, és bármilyen furcsán hangzik is, szerintem csak a lemez lehet a bűnös, ugyanis ugyanezen időszakban is ugyanazon a gépi konfiguráción más típusú (SENTINEL) lemezzel soha nem tapasztaltam ezt.** Azt sem tartom valószínűnek, hogy téves beütés okozta a hibát, – BASF lemez esetén eltévesztem, a SENTINEL esetében pedig soha nem tévesztem el?

Véleményem szerint igen kíméletes módon bánok a lemezeimmel, ennek ellenére a BASF lemezeim élettartama igen rövid, hamar tönkremennek. A lemez a befogási helyén nem rendelkezik merevítő gyűrűvel, egy idő után itt megnyúlik és előbb-utóbb olvashatatlanná válik az egész lemez (már a tartalomjegyzék is – esetleg néhány blokk direkt módon elérhető).

Programjaimban általában direkt adattárolást használok, ezért adatlemezeimet használat előtt letesztelem, a lemez valamennyi blokkját formatálás után 2-szeri teleírással és visszaolvasással ellenőrzöm. Vadonatúj BASF és WABASH lemezek között találtam olyanokat, melyen valamelyik blokk hibás (nem írható/olvasható). Az adott lemezen a hiba bármikor reprodukálható volt más módon is. (Megjegyzem, nem sok WABASH lemezt használtam, így nem tudom, mennyire általánosítható tapasztalatom erre a típusra.)

A másik két típussal még nem volt rossz tapasztalatom. Van 3 éve rendszeresen nyúzott SENTINEL lemezem, még jó. Sajnos, mostanában ez a típus nem kapható. Kb. fél éve PARROT lemezeket használok, még nem találtam hibását.

Bacsur Kálmán
2921 Komárom,
Gesztenye u. 1. fsz. 2.

APRÓCSKÁK

VC 20

VC 20-al rendelkezők ismeretségét keresem információ-, tapasztalat-, illetve programcsere céljából.
Jaksa János, 1116 Budapest, Ezüstfenyő tér 7.

C 16, Plusz 4

Orosz karakteres készletet keresek bővítős Commodore 16-os gépre. Lehetőleg szubrutinként kezelhetőt.
Szűcs Pál, 1211 Budapest, Táncsics M. u. 85.

Commodore 16-os cserepartnert keresek. Program, leírás, hardver, angol nyelvű újságok.
Szűcs Pál, 1211 Budapest, Táncsics M. u. 85.

Angol és orosz nyelvoktató programok eladók (Plusz 4, C 16, C 116).
Kálmán Albert, 3300 Eger, Rákóczi út 31. III/11. Tel. üzenet: 143-031, 330-345 (Bp.)

Commodore 16-os, 116-os programokat cserélek. Lehetőleg programlistát kérek!
Modla Norbert, 7622 Pécs, Bajcsy Zs. u. 3. Tel.: 100-62

Plusz 4-es játék- és egyéb programokat cserélek.
Varga Sándor, 1213 Budapest, Erdősor u. 165/A. IV/19.

Általános Iskola technika tantárgyának tanítását segítő C 16-on és Plusz 4-en futtatható, kazettán tárolt programokat cserélnénk.
Központi Műhely, 7800 Siklós, Felszabadulás u. 62.

C 16-os programokat cserélek és felajánlok fordítás céljából egy német nyelvű „COMMODORE BEDIENUNGSHANDBUCH" c. könyvet, mely gépi kóddal is foglalkozik C 16-on.
Jászvári Péter, 8200 Veszprém, Koltói út 1/B.

C 16 és Plusz 4-es játék- és felhasználói programokat cserélek. Keresek C 16-os compilert.
Süli Z. Rudolf, 6750 Algyő, Téglás u. 101.

C 16-os programokat cserélnék.
Rittgasser István, 7632 Pécs, Ilku P. 25. III/10.

Vennék Datasette 1531 típusú (C 16-hoz való) magnót. Árajánlatot kérek.
Pintácsi Péter, 5000 Szolnok, Orosz Gy. út 26. Tel.: (56)12-378 délután

---

Plusz 4-re vagy C 16-ra gépi kódú programozás elsajátításához keresek magyar, esetleg angol nyelvű dokumentációt.
Igali István, 8418 Bakonyoszlop, Kossuth u. 2/1.

C 64

C 64-es felhasználói és játékprogramokat cserélek kazettán.
Nagy Imre, 6100 Kiskunfélegyháza, Kálvária u. 16.

C 64 könyvek eladók: Becker: Gépi kód C 64-en; Tippek, trükkök; VC–1541 programozása; BASIC programozás magasiskolája; 64 intern; 1001 játék; Felhasználói kézikönyv; Halász: Alapismeretek; Úry: C 64 I–II. Külön-külön is.
Csiszár Zoltán, 8000 Székesfehérvár, Lövölde u. 19. V/3.

C 64-re keresek padlófűtés tervező programot saját padlófűtésem megtervezéséhez.
Kováts Gábor, 1185 Budapest, Nagybecskerek u. 30.

C 64-en futó, gyári vagy saját készítésű játékprogramokat vennék kazettán.
Bartha György, 5000 Szolnok, Köztársaság út 18.

Adatbáziskezelő programok elkészítését olcsón vállalom C 64-re.
ifj. Hódosi Benjámin, 2151 Fót, Madách u. 56.

C 64-es játékprogramokat cserélek. Repülés és más szimulátorok is érdekelnek.
Kulcsár Péter, 2400 Dunaújváros, Vöröshadsereg út 160.

C 64-es játék- és egyéb programokat cserélek kazettán.
ifj. Kiss János, 6800 Hódmezővásárhely, Árpád u. 2.

C 64-es játékprogramokat cserélek floppy disken.
Harcsa Béla, 1025 Budapest, Törökvész u. 131/b.

C 64-es játék- és felhasználói programokat cserélek kazettán. Programleírások is érdekelnek. Listát kérek.
Molnár Zoltán, 9200 Mosonmagyaróvár, Gazdász u. 10.

Vásárolnék C 64-hez hibás Datasette-et.
Knyur Géza, 6800 Hódmezővásárhely, Gellért u. 9.

VEGYES

Commodore és Sinclair gépek, perifériák javítását vállaljuk. Közületek részére átalánydíjas karbantartási szerződéssel is rendelkezésére állunk. Érdeklődni: Elektróra GMK. 4025 Debrecen, Nyugati u. 2.

Fotoelektronik–Novotrade GT. értesíti tisztelt ügyfeleit, hogy 1986. december 11-től csütörtöki napokon meghosszabbított ügyfélszolgálatot tart 17 óra 30 percig Magyar u. 12–14. sz. alatti szervizében.

MPS 802 printer eladó.
Kiskuti Gábor, Tel.: 690-834, 8–15 óráig.

GEOS

1. rész

Az 1986-ban megjelent operációs rendszer használható valamennyi C 64-es gépen. Az általunk közölt leírás nem azonos az eredeti „Használati útmutató"-val, nem annak fordítása, hanem HONTI TAMÁS több hónapos kísérleteinek, gyakorlatának eredménye. A leírás a gyári GEOS-szal dolgozóknak nyújt segítséget, a kalózmásolatok alkalmazásának trükkjeit nem tartalmazza.

A GEOS BEJELENTKEZIK

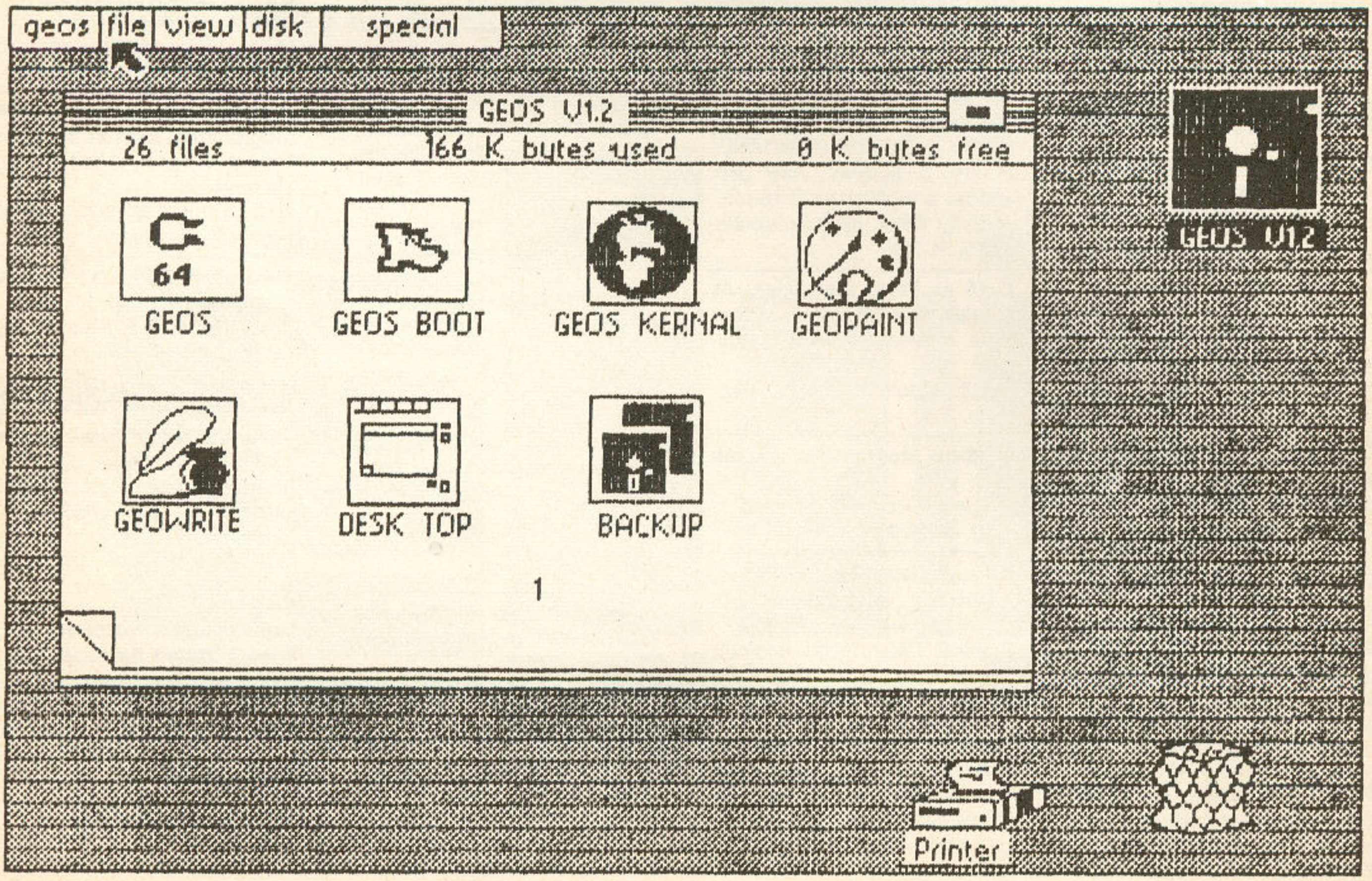
Dugjuk a joysticket a PORT 1-be és töltsük be a programot [LOAD"GEOS", 8, 1]. A képernyőn a bejelentkezés után az ún. munkaasztalt (DESK TOP) látjuk. Ezen a munkaasztalon fogunk a GEOS-szal dolgozni. Munkaeszközünk a joystick, amellyel egy kis nyilacskát mozgathatunk a képernyőn. Ezzel a nyilacskával választhatjuk ki a különböző funkciókat, vagy parancsokat adhatunk a segítségével. A képernyőn látható különböző ábrák és szövegek értelmezéséről később lesz szó.

Ha vásároltunk, vagy az új gépünkhöz kaptunk egy GEOS program-csomagot, akkor az első dolgunk az legyen, hogy készítsünk róla azonnal egy biztonsági másolatot, majd legalább két munkakópiát.

Hogy ennek mi a módja, arról a biztonsági másolat készítése című fejezetben olvashatnak.

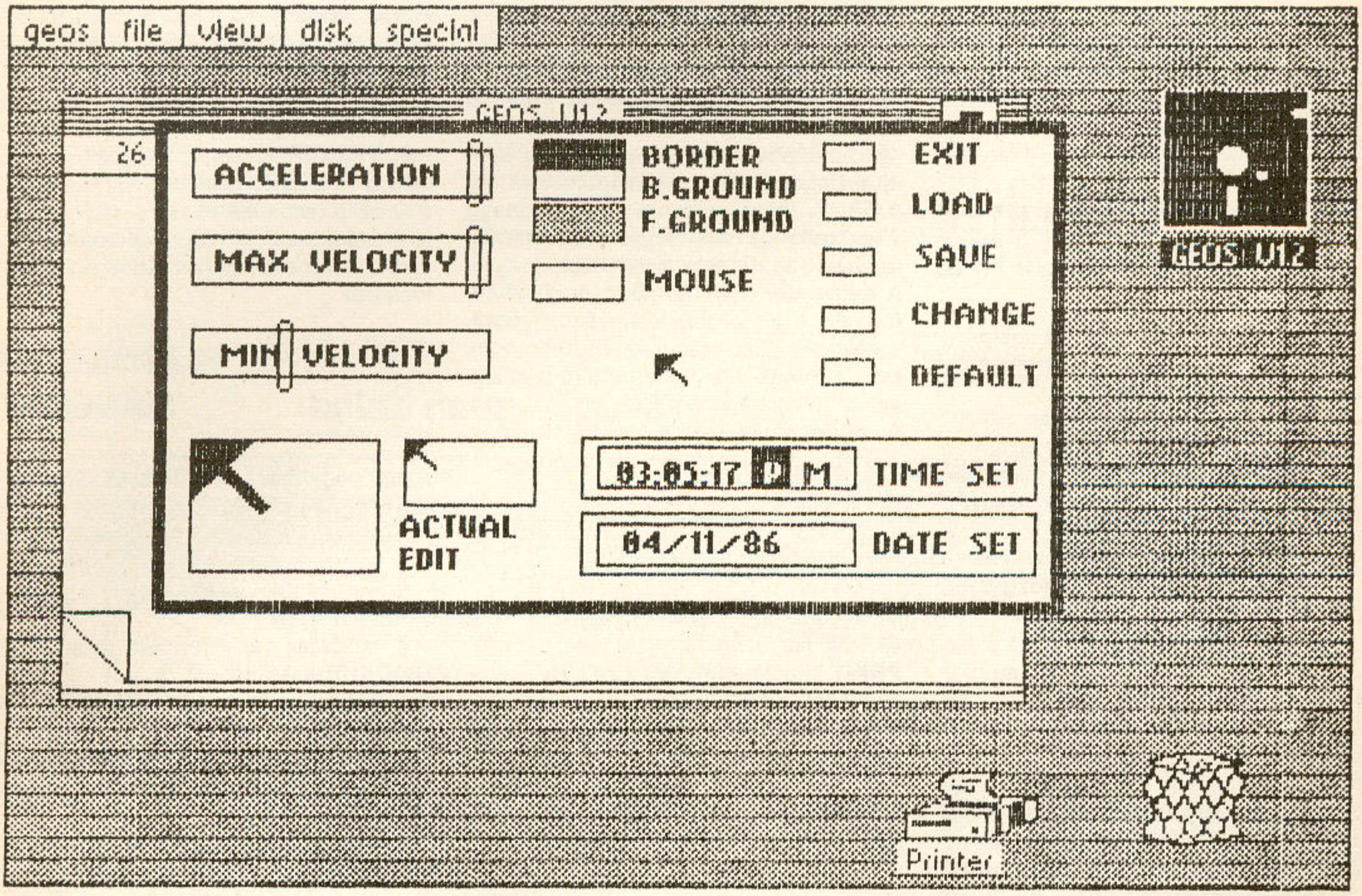
A GEOS KEZELÉSÉRŐL

A GEOS egyszerűen kezelhető rendszer. Az ismerkedést kezdjük a munkaasztallal (DESK TOP). Az alábbi rövid magyarázattal ellátott ábrán tanulmányozhatjuk a GEOS munkaasztalát:



A munkaasztalhoz tartozó legfontosabb munkaeszköz a nyilacska. A felhasználó a képernyőn a joystick által mozgatott nyilacska segítségével kiválaszthatja a parancssorból valamelyik menüpontot és a tűzgomb lenyomásával aktivizálhatja azt. Bármelyik parancs aktivizálása után egy almenübe kerülünk, s itt további választási lehetőségünk van. (Almenü = PULL DOWN MENU)

Vegyünk egy konkrét példát a fentiek jobb megértéséhez.
Válasszuk ki a parancssorból a **GEOS** parancsot, majd az almenüből a **PREFERENCE MGR**-t aktivizáljuk. A képernyőn megjelenő kommunikációs ablakon állítsuk be az aktuális időpontot: **TIME SET.** Először ráállunk a nyilacskával a **TIME SET**-re, majd megadjuk, hogy délelőtt, vagy délután van (A vagy P). Utána bebillentyűzzük a pontos időt, (SPACE-szel mozoghatunk). Vegyük figyelembe, hogy az óra 12 órás ciklusokat mér! Most már csak az akutális dátumot – **DATA SET** – kell beállítani. (Amerikai rendszerű a dátum, tehát: Hó/Nap/Év→ 06/30/86). Az eljárás azonos a TIME SET-nél leírtakkal. Az **ACCELERATION** menüponttal a nyilacska gyorsulását, a MAX-, ill. MIN **VELOCITY** menüpontokkal a nyilacska maximális illetve minimális sebességét szabályozhatjuk. A szerkesztő mezőben levő nyilacskát tetszésünk szerint átrajzolhatjuk, majd az előtér **(F. GROUND),** a háttér **(B. GROUND),** a keret **(BORDER)** és a nyilacska **(MOUSE)** színét is beállíthatjuk. Összesen 16 szín áll a rendelkezésünkre. Mindezt egyszerű joystick mozgatással és a tűzgomb szükség szerinti használatával. A beállított paramétereket lemezre menthetjük **(SAVE)** vagy onnan betölthetjük **(LOAD),** aktivizálhatjuk **(CHANGE),** vagy az eredeti értékeket visszaállíthatjuk **(DEFAULT).** Ha nem akarjuk a változtatásokat letárolni, csak az éppen végzett munkánál kívánunk az újonnan beállított paraméterekkel dolgozni, akkor a **CHANGE** menüpontot kell a kilépés előtt **(EXIT)** választani. MEGJEGYZÉS: Az amerikai és európai hálózati frekvencia eltérése miatt (60 Hz, ill. 50 Hz) az óra sokat késik. Az alábbi ábrán tanulmányozhatjuk az elmondottakat.



A PIKTOGRAMOK

A GEOS bejelentkező képernyőjén apró képecskéket találunk. Ezeket a rendszer ICON-oknak nevezi. Mi a nálunk jobban elterjedt piktogram kifejezést fogjuk használni. (A piktogramokat a GEOS több célra használja.) A piktogramok a lemezen található file-okat jelképezik.
A GEOS az alábbi file típusokat különbözteti meg:
– Rendszer adat-file-ok, ezek a GEOS rendszer alapelemei
– Program adat-file-ok, ezek a GEOS alatt írt programok, pl. a GEOPAINT vagy a GEOWRITE továbbá assembly és BASIC programok. Ezeket az egyszerűség kedvéért felhasználói programoknak fogjuk nevezni.
– Adat file-ok, ezek olyan munka file-ok, amelyeket mi hoztunk létre
– I/O vezérlő programok, joystick, egér, különböző nyomtatók használatához
– Nem GEOS file-ok, ezek tulajdonképpen a hagyományos C–64-es file-ok.
A piktogramok kiválasztásával és aktivizálásával végezhetők el a különböző műveletek. **Ha meg akarjuk tudni pl.: milyen programok találhatók egy másik lemezen, akkor elég ha a vizsgálandó lemezt betesszük a meghajtóba, majd a jobb felső sarokban található – lemezt szimbolizáló – piktogramra visszük a nyilacskát és tüzet nyomunk.** Ha a vizsgált lemez GEOS rendszerű, akkor azonnal megjelenik a lemeztartalom a képernyőn. Ha nem GEOS rendszerű a lemez, akkor az alábbi kérdésre kell választ adni:
This is a NON-GEOS disk.
Would you like it converted?
YES NO
Magyarul: Ez nem GEOS rendszerű lemez.
Kéri-e a konvertálást (átalakítást)?
Igen Nem

A NO válasz esetén – a lemez formájának és tartalmának érintetlenül hagyása mellett – megjelenik a lemeztartalom a képernyőn. Ilyenkor a file-okat egy-egy 'C–64' piktogram jelképezi, alattuk a file-ok nevével. Ha kiválasztjuk és aktivizáljuk közülük valamelyik program file-t akkor azt a GEOS rendszer betölti és elindítja.

Sajnos YES válasz esetén sem történik szemmel látható változás a lemezünkkel, csupán annyi a különbség, hogy ezután az átalakítás után nem lesz idegen a GEOS számára ez a lemez. (Valóságban annyi történik, hogy a lemezen megfelelő területeket lefoglalja a rendszer magának. Ezeken a területeken tárolódnak az egyes file-ok információi, és a keretre kitett file-ok jellemzői.)

Ha lapozni akarunk a tartalomjegyzékben, akkor a bal alsó sarokban található, visszahajtott papírsarok szimbólumra állunk rá és tűzgombot nyomunk. Lapozhatunk úgy is, hogy a billentyűzeten a kiválasztott oldal számát beütjük.

MŰVELETEK A PIKTOGRAMOKKAL

– **kiválasztás:** a piktogramra állítjuk a nyilacskát és tüzet nyomunk, hatására a piktogram invertálódik.
– **mozgatás:** a kiválasztás után néhány mp-ig várakozunk, majd ismét tüzet nyomunk, hatására megjelenik a piktogram szellemképe, amit tetszés szerint mozgathatunk
– **aktivizálás:** ha várakozás nélkül nyomjuk meg másodszor is a tűzgombot a kijelölt program file aktivizálódik
– **áthelyezés:** a kiválasztott majd aktivizált piktogram áthelyezhető – a munkaterületen kívülre,
– másik oldalra,
– a printerre,
– a szemétkosárra.

Fentieken túl végezhetők ún. kombinált műveletek is. A kombinált műveletek során a piktogramokkal és a parancssor valamelyik menüpontjával együtt dolgozunk. A kombinált műveletekre jó példa lesz a munkalemez elkészítése.

A PARANCSSOR

Térjünk vissza a parancssorhoz. Öt különböző menüpontot tartalmaz, az alábbiak szerint: GEOS, FILE, VIEW, DISK, SPECIAL

GEOS

A **GEOS info** tájékoztat a készítőkről és arról, hogy hányadik változat van a birtokunkban.

A **deskTop info** a rendszer készítésével kapcsolatos, a felhasználó szempontjából lényegtelen információkat tartalmaz.

A segédprogramok – **DESK ACCESSORIES** – közül első a **note pad.** Tulajdonképpen egy notesz, ami a program futása során bármikor meghívható és abba a felhasználó a fontosabb megjegyzéseit beleírhatja.

A **preference mgr** menüpont aktivizálása után beállíthatjuk a színeket, az órát és a dátumot. Tetszésünk szerint átrajzolhatjuk a nyilacska formáját és módosíthatjuk az egér munkaparamétereit. Természetesen a megváltoztatott állapot lemezre menthető és legközelebb már az általunk beállított értékekkel indul a program.

A **photo manager/photo album** a rendszer mellékszolgáltatása. Azokat a képeket tárolhatjuk el, amelyekre később is szükségünk lehet.

A **select printer** menüpont segítségével az aktuális printerhez illeszthetjük a rendszert. A printerek választékára igazán nem lehet panasz. A nagyobb cégek által gyártott printerekhez jól illeszthető a GEOS.

A **text manager** segítségével a fontosabb szövegek tárolhatók a lemezen.

A **calculator** egy bármikor meghívható funkció. Egyszerűbb számítások végezhetők vele. Közönséges zsebszámológép módjára működik. Használható joystickkel vagy billentyűzetről.

Az **alarm clock,** mint a nevéből is kitűnik egy ébresztőóra.

FILE

OPEN file megnyitása
CLOSE file lezárása
PRINT file nyomtatása
RENAME file átnevezése
DUPLICATE file megduplázása
INFO a kiválasztott file fontosabb tulajdonságait tudjuk meg. Itt van lehetőség a törlés elleni védelem megszüntetésére, vagy a védelem beállítására.

VIEW

A lemeztartalom kijelzésének megváltoztatására szolgál. Alapállapotban a lemeztartalom piktogram **(ICON)** formájában jelenik meg és ebben a formában aktivizálhatók az egyes programrészek. Lekérdezhető viszont a lemeztartalom a program típusa **(TYPE),** hossza **(SIZE),** készítési ideje **(DATE)** és ABC sorrend **(NAME)** szerint is.

DISK

Ezen menüpont alatt találjuk meg összegyűjtve a lemezműveletek elvégzéséhez szükséges utasításokat. Itt tudunk lemezt formattálni **(FORMAT),** átnevezni **(RENAME),** másolni **(COPY),** validálni **(VALIDATE),** a lemezt megnyitni **(OPEN)** vagy lezárni **(CLOSE)**... stb.

FONTOS FIGYELMEZTETÉS! Ha egy lemez meg van nyitva, soha ne vegyük ki a meghajtóból. A kivétel előtt a lemezt mindig le kell zárni (CLOSE) vagy a jobb felső sarokban található kilépő mezőn keresztül kell kilépni.

A GEOS lemezt nem célszerű BASIC-ből kezelni. A rendszer nagyon érzékeny az ilyesmire. Ha mégis megpróbáljuk annak csak egy eredménye lesz, hogy a rendszer lemerevedik.

SPECIAL

Három lehetőséget kínál: **RESET, BASIC** és **Q-LINK.** A RESET funkció lehetőség a program megszakítására és újratöltésére. A BASIC funkció a BASIC-be való visszatérést szolgálja. Ha a BASIC után ismét a GEOST kívánjuk használni, akkor elég a RESTORE gombot megnyomni.

A Q-LINK nem más, mint egy csatlakozási lehetőség az USA-ban működő QUANTUM-LINK nevű telekommunikációs hálózathoz.

BIZTONSÁGI MÁSOLAT

Most, hogy már ismerjük a GEOS alapelemeit, nekiállhatunk a biztonsági másolat elkészítésének. Két lehetőségünk van:
– fizikai másolóval átmásoljuk a lemeztartalmat, vagy
– a rendszer saját másoló programját (BACKUP) használjuk.

Az utóbbi esetben az alábbiak szerint kell eljárni:
– A munkaasztalról **(DESK TOP)** jelöljük
– nyomjuk meg egyszer a tűzgombot, majd
– mozgassuk a nyilacskát a parancssorra és válasszuk ki a **FILE** parancsot (tűzgomb!)
– a leguruló almenüből válasszuk ki a tűzgomb ismételt lenyomásával az **OPEN** menüpontot.

A következő üzenetet fogjuk kapni:
DISK BACKUP/RESTORE UTILITY
INSERT DESTINATION DISK TO BE FORMATTED
AND ENTER 'F' TO FORMAT OR 'Q' TO QUIT (F/Q)

Magyarul: formattáláshoz nyomjon 'F'-et, vagy 'Q'-t a kilépéshez.

Az 'F' billentyű lenyomásával megindul a kicsit hosszadalmasnak tűnő másolás.

A továbbiakban a copy üzenetei alapján kell eljárni.

FORMATTING DESTINATION DISK...	folyik a céllemez formattálása,
INSERT SOURCE DISK, AND ENTER 'C' TO COPY (C)?...	helyezze be a forráslemezt és nyomjon 'C'-t a másolás folytatásához,
READING SOURCE DISK...	forráslemez olvasása,
PLEASE INSERT DESTINATION DISK	helyezze be a céllemezt,
WRITING DESTINATION DISK...	folyik a céllemez írása,
PLEASE INSERT SOURCE DISK	helyezze be a forráslemezt,
BACKUP COMPLETE!...	kész a másolás,
INSERT GEOS BOOT DISK AND PRESS RESTORE	tegye be a GEOS lemezt és nyomjon [RESTORE]-t

Természetesen az 'eredeti' GEOS lemezről van szó, mivel a GEOS saját másoló programjával (BACKUP) készült másolat csak és kizárólag a rendszerlemez előzetes betöltése után működik! Ezért célszerűbb (bár nem jogszerű) egy jól ismert fizikai másoló használata!

Ha kész a másolás, akkor próbáljuk ki az új lemezünket. A GEOS-hoz való visszatéréshez tegyük be az eredeti lemezt és elég a RESTORE gombot megnyomni, a rendszer automatikusan be fog töltődni. Az indítás után már fut a munkalemezen is a GEOS.

Ha elkészült a biztonsági másolat és a programot használni is akarjuk, akkor legalább 2 munkalemezt kell csinálni.

MUNKALEMEZ ELKÉSZÍTÉSE

A fent leírtak szerint készítsünk két lemezmásolatot, ebből fogjuk az alábbi eljárással a két munkalemezt elkészíteni.

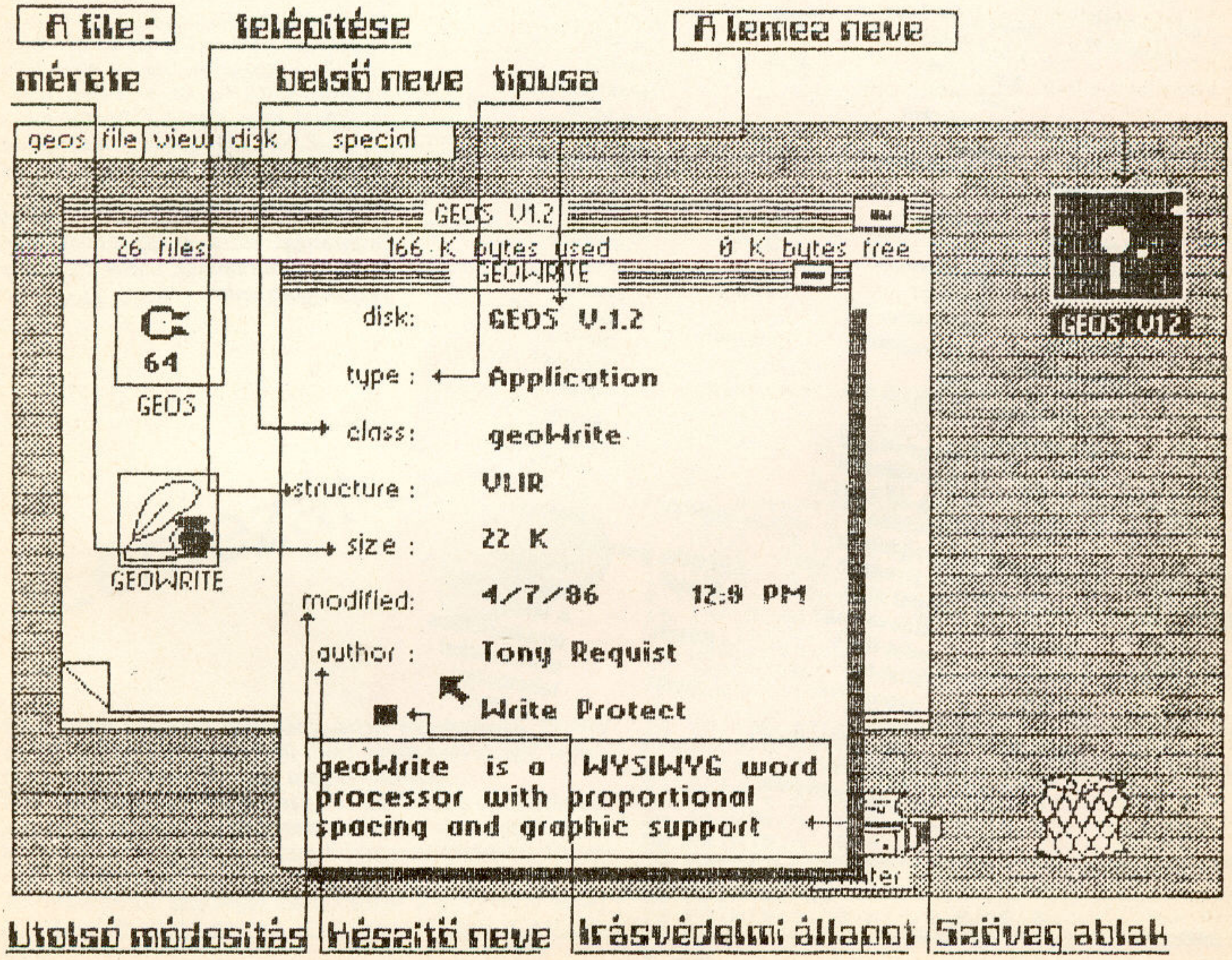
Először is adjunk új nevet a munkalemezünknek. (DISK parancs RENAME utasítás.)

Arra mindig gondosan ügyeljünk, hogy két azonos nevű lemezünk soha ne legyen! Sok kellemetlenségtől kíméljük meg ezáltal magunkat.

A következő feladat a rendszerlemez átalakítása munkalemezzé. Ehhez a fölösleges file-okat le kell törölni a lemezről. Ha a munkalemezt grafikus feladatok megoldására akarjuk használni, akkor nem lesz szükség pl.: a GEOWRITE-ra. Letörlésével nagy memória területet szabadíthatunk fel. Ha nagyobb a memória igényünk, mint amennyit így nyertünk, akkor nyugodtan letörölhetjük pl. azokat a betűtípusokat (FONTS), amelyeket nem kívánunk használni. A nyomtatók közül is csak azt érdemes a munkalemezen hagyni, amelyikkel majd nyomtatni fogunk.

A törlés módja: a már ismertetett módon kijelöljük a törölni kívánt **GEOWRITE** file-t, majd ezt követően a parancssorból kiválasztjuk a **FILE** parancsot. Ezután az almenüből az **INFO** utasítást kell kiválasztani.

Keressük meg a képernyőn az írásvédelem **(WRITE PROTECT)** felirat előtt elhelyezkedő fekete négyzetet. A nyilacska és a tűzgomb használatával szüntessük meg az írásvédelmet és térjünk vissza a kilépési ponton keresztül (jobb felső sarok) a munkaasztalra. A törlés – a védelem megszüntetése miatt – már két lépésben elvégezhető. Először kijelöljük a törölni kívánt **GEOWRITE** file-t. Ekkor a



piktogram invertálódik. Néhány másodperc várakozás után a tűzgomb ismételt megnyomásának hatására megjelenik az invertálódott piktogram szellemképe, amit a joystickkal a szemétkosárhoz mozgatunk. A tűzgomb újabb megnyomására a rendszer a kijelölt file-t letörli a lemezről.
Így rendelkezésünkre áll egy GEOPRINT munkalemez, amely a GEOS-rendszer összes előnyeivel rendelkezik.

FILE-OK MÁSOLÁSA

Ha már a másolásoknál tartunk, érdemes megnéznünk milyen sajátos módon történik a file-ok egyik lemezről a másikra való másolása:

– először kijelöljük az ismert módon a másolni kívánt file-t és azt a munkaterületen kívülre helyezzük,
– második lépésként betesszük a meghajtóba a céllemezt és kiválasztjuk a lemezt szimbolizáló piktogramot. Hatására be fog töltődni a másik lemez tartalomjegyzéke,
– a munkaterületre helyezzük a másolni kívánt file-t.
A program kérni fogja a forráslemezt, majd a céllemezt (esetleg többször is egymás után) és automatikusan elvégzi a file másolását.
A file-ok nemcsak másolhatók, hanem át is rendezhetők a lemezen. Ha a lapon belül akarunk sorrendet változtatni, akkor a megcserélendő file-okat a munkaterületen kívülre tesszük, majd a kívánt sorrendbe visszarakjuk.

Hasonló módszerrel a másik oldalra is átvihetők a file-ok.
A törlés a szemétkosáron keresztül történik. Ha a file írásvédett, akkor a törlési szándékot követően az alábbi üzenetet kapjuk:
„THIS FILE IS WRITE PROTECTED AND CAN'T BE DELETED."
Magyarul: A file írásvédett, nem törölhető.
Ilyenkor először meg kell szüntetni a már ismert módon az írásvédelmet.

EGYÉB TUDNIVALÓK

Két meghajtó egység használata többféleképpen is lehetséges. Ha a második meghajtó hardver-száma 9-es, akkor azt a GEOS felismeri.
Ha mindkét meghajtó 8-as, akkor a következők szerint járunk el. A munka megkezdésekor csak az egyik meghajtót kapcsoljuk be, majd töltsük be a GEOS rendszert. A bejelentkezés után válasszuk ki a DISK menüpontból az ADD DRIVE utasítást, majd a kommunikációs ablakban megjelenő üzenetek szerint kell eljárni. Az elsőként bekapcsolt meghajtó lesz a 8-as vagy 'A', a másodikként bekapcsolt meghajtó pedig a 9-es vagy 'B' egység. A két meghajtóban levő lemezek közül egyszerre csak az egyik lehet megnyitva; mindig az, amelyiket a nyilacskával és a tűzgomb használatával megnyitottuk. Bármelyiket megnyithatjuk, a GEOS a másikat automatikusan lezárja!
Legközelebbi számunkban a GEOS programjai közül a GEOPAINT nevű grafikai programmal, annak használatával foglalkozunk.

GEOS parancs:	
– Geos info	tájékoztat a készítőkről és a változat számáról
– DesKTop info	verziószám
– Note Pad	notesz
– Preference mgr	preferenciák
– Photo mgr	fotoalbum
– Select Printer	illeszti a kiválasztott printert
– Text mgr	szövegalbum
– Calculator	zsebszámológép
– Alarm clock	ébresztő óra
FILE parancs:	
– Open	megnyit egy felhasználói prg.-ot vagy adat file-t
– Duplicate	megdupláz a lemezen egy adat file-t
– Rename	átnevez egy adat-file-t
– Print	adat-file nyomtatás
VIEW parancs:	
– Icon	piktogram (icon) szerinti lemeztartalom
– Name	név szerinti lemeztartalom
– Date	dátum szerinti lemeztartalom
– Size	file nagyság szerinti lemeztartalom
– Type	file típus szerinti lemeztartalom
DISK parancs:	
– Open	megnyit egy lemezt
– Close	lezár egy lemezt
– Rename	átnevez egy lemezt
– Copy	átmásol egy lemezt
– Add drive	átszámozza a lemez meghajtót
– Validate	átszervez egy lemezt
SPECIAL parancs:	
– Basic	visszatér a BASIC-be
– Reset	a lemez újrastartolása
– Qlink	csatlakozási lehetőség a Q-LINK hálózathoz.

HIPERKOCKA

A Tudomány 1986/6 számában olvashattuk A. K. Dewdney hiperkocka programját.

Röviden a lényeg: egy négydimenziós kockát forgathatunk különböző dimenziósíkok körül, és a forgatás után a kocka háromdimenziós vetületét jeleníthetjük meg.

Először a forgatás szögét kell megadnunk, majd a lehetséges forgatási síkok kiírását láthatjuk. A kurzort léptessük a kívánt sík sorszámára és nyomjunk RETURN-t. Ha további sík körül is akarunk forgatni, a továbbforgatom kérdésre igennel válaszoljunk, és újra a forgatási szög kérdezéséhez jutunk. Ha válaszunk nem, megjelenik a kocka képe. Gombnyomásra újra indul a program.

A rajzolás SIMON'S BASIC segítségével történik.

Más segédprogram esetén a 270, 280 sorok utasításait kell a megfelelőre kicserélni. ... A 110–130 sorokat pontosan gépeljük be, különben rossz lesz az értékadás.

Igazán akkor lesz szép a program, ha színezni is tudjuk a lapokat. (Bizonyos forgatásoknál akár 12 lap is látszódhat.) Akinek ez sikerül, még a négydimenziós láttatással is megpróbálkozhat. A megvalósítás nagyon egyszerű, csak a 64-es lehetőségei korlátozottak.

Jó próbálkozást! **Allaga Gyula**

C= 64

```
1 REM ***********************************
2 REM *      C= UJSAG SORSZAM 046        *
3 REM *      HIPERKOCKA  (SIMON'S)       *
5 REM *      PROGRAM ALLAGA GYULA        *
6 REM *      1028 BP JAROKA U. 8         *
7 REM ***********************************
10 REM***** ADATOK *****
20 DIMCS(16,4),TA(16,4),RO(4,4),VO(33):X=20
30 FORI=1TO16:A=I:FORJ=4TO1STEP-1
40 IF(A-1)/(2↑(J-1))>=1THENCS(I,J)=10:A=A-2↑(J-1):GOTO60
50 CS(I,J)=-10
60 NEXTJ,I:FORI=1TO33:READVO(I):NEXTI
70 DATA1,2,4,3,7,15,11,9,10,12,4,8,16,15,13,14,10
80 DATA2,6,8,7,5,13,9,1,5,6,14,16,12,11,3,1
90 REM***** MERRE ? *****
100 PRINT"□":INPUT"A FORGATAS SZOGE (FOK)";C
110 PRINT"□▮▮1 ROT 1-2":PRINT"▮▮2 ROT 1-3"
120 PRINT"▮▮3 ROT 1-4":PRINT"▮▮4 ROT 2-3"
130 PRINT"▮▮5 ROT 2-4":PRINT"▮▮6 ROT 3-4"
140 INPUT"▮LEPJEN A MEGFELELO▮SORSZAMRA+RETURN▮▮▮▮";D$
150 A=VAL(MID$(D$,7,1)):B=VAL(MID$(D$,9,1))
160 D=VAL(LEFT$(D$,2)):IFD<1ORD>6THEN110
170 REM***** SZAMOLS *****
180 E=COS(C*π/180):F=SIN(C*π/180)
190 FORI=1TO4:RO(I,I)=1:NEXTI
200 RO(A,A)=E:RO(B,B)=E:RO(A,B)=F:RO(B,A)=-F
210 FORI=1TO16:FORJ=1TO4:FORK=1TO4
220 TA(I,J)=CS(I,K)*RO(K,J)+TA(I,J):NEXTK,J,I
230 REM***** KOORD.TRANSZ. *****
235 GOTO300
240 FORI=1TO16:FORJ=1TO2
250 TA(I,J)=X*TA(I,J)/(X-TA(I,3)):NEXTJ,I
260 REM***** RAJZ *****
270 HIRES0,1
275 FORI=1TO32
280 LINE TA(VO(I),1)+160,TA(VO(I),2)+100,TA(VO(I+1),1)+160,TA(VO(I+1),2)+100,1
285 NEXTI
290 GETS$:IFS$=""THEN290
295 NRM :PRINT"□";RUN
300 PRINT"□TOVABB FORGATOM (I/N)"
310 GETA$:IFA$=""THEN310
320 IFA$="N"THEN240
330 IFA$<>"I"THEN310
340 FORI=1TO16:FORJ=1TO4
350 CS(I,J)=TA(I,J):TA(I,J)=0:NEXTJ,I
360 FORI=1TO4:FORJ=1TO4
370 RO(I,J)=0:NEXTJ,I:GOTO90
```

Lemezátnevező

C= 64

Ez a rövid program leginkább akkor használható, amikor valaki a lemeztárát rendezi, mert segítségével egy lemez neve átírható anélkül, hogy megformáznánk.

Ezen kívül még két szolgáltatása van, ami inkább csak érdekesség: lehetővé teszi, hogy a tartalomjegyzéket eltüntessük, illetve előhívjuk, valamint, hogy a tartalomjegyzék kiírásakor képernyőt töröljön a gép.

Az utóbbiakhoz szükséges parancsokat a program a lemez fejlécébe rejti, ezért ilyenkor a lehetséges névhossz 16-nál rövidebb.

```
0 REM *****************************
1 REM *     C= UJSAG   SORSZAM 044  *
2 REM *     LEMEZNEV                *
3 REM *     PROGRAM: TOROK AKOS     *
4 REM *     2500 ESZTERGOM          *
5 REM *     BOTTYAN J. U. 10        *
6 REM *****************************
7 POKE53280,0:POKE53281,0:PRINT"▮"
8 INPUT "□KIIRJA A DIREKTORYT (I/N)";I$:
IFI$<>"I"ANDI$<>"N"THEN8
10 INPUT"▮LEGYEN KEPERNYOTORLES (I/N)";C
$:IFC$<>"I"ANDC$<>"N"THEN10
15 INPUT"▮A NEVE (MAX.16 BETU) ";N$:IFLE
N(N$)>16THEN15
20 N$=LEFT$(N$+"                ",16)
21 INPUT"▮AZ ID  (MAX. 5 BETU) ";ID$:IFL
EN(ID$)>5THEN21
22 ID$=LEFT$(ID$+"     ",5)
23 IFI$="N"THENID$=LEFT$(ID$,2)+CHR$(0)+
CHR$(0)+CHR$(0)
24 IFC$="I"THENN$=CHR$(13)+CHR$(147)+CHR
$(18)+CHR$(34)+LEFT$(N$,12)
30 OPEN15,8,15:OPEN1,8,2,"#"
40 PRINT#15,"B-R:";2;0;18;0
50 PRINT#15,"B-P:";2;144
55 K$=N$+CHR$(160)+CHR$(160)+ID$+CHR$(16
0):PRINT"▮"K$
60 PRINT#1,K$
70 PRINT#15,"B-W:";2;0;18;0
80 CLOSE1:CLOSE15:OPEN15,8,15,"I":CLOSE1
5:END
```

ifj. Gulyás László:

ASSEMBLER

A C 16-os, Plusz 4-es gépcsalád rendelkezik beépített monitorral, ma már ezt így illik. Csekélyke méretei miatt azonban inkább csak hibakeresésre, javításra való, kevésbé alkalmas programok írására, fejlesztésére. Ezért írtam ezt a programot, amelyben használhatunk címkéket, megjegyzéseket, „.BY"-vel vihetünk be adatokat, és a Plusz/4-es (64K-s C–16) gépnél TEXT utasítást is használhatunk. A program elég gyorsan végzi a fordítást, annak ellenére, hogy BASIC-ben íródott. Futtatás után „ASSEMBLER" felírást kapunk, majd megjelenik egy „?", ilyenkor a program várja az utasítást.*

Parancsok

A: Hexa-dec. ill. dec.-hexa átszámítás. A kérdőjelre kell beírni az átszámítandó számot. Ha ez hexa, akkor „$"-t kell írni a szám elé.

D: Sortörlés. (DELETE) Kezdőcímtől végcímig kitörli a forrásprogram sorait, majd meghívja a listázó rutint.

E: Kiírja az object program elejét és végét (decimálisan).

F: Fordítás. Az „F.C." kérdésre azzal a memóriacímmel kell felelni, ahova a programot fordítani kívánjuk. A „T.C." kérdésre, azzal a memóriacímmel kell felelni, ahová az object programot tölteni kívánjuk. Mindkét címet decimálisan kell megadni. Ajánlott címek:
16K-n: 1440–2047
64K-n: 24576-(PS-170)

G: A megadott címtől futtatja az object programot (a címet decimálisan várja).

I: A forrásprogram beírása. Ha az „AUTO?" kérdésre „I"-vel felelünk, akkor meg kell adni a kezdő sorszámot és a lépésközt, a „K.LE;?" kérdésre. Ezután írhatjuk a forrásprogramot. Befejezés: „."-tal (ponttal).
Ha „N"-nel válaszolunk, akkor a következőképpen írhatjuk a programot:
sorszám, utasítás. Pl. **10,LDA#$FF**
Befejezés:
sorszám,.(pont) Pl. **11,.**
Maximális sorszám 16K-n: 50, 64K-n a 60-ik sorban a PS változó állítja. Alapállapotban: 170.

J: Javítás. A javítandó sor környezetét listázza (előtte 10 sor, utána 10 sor). A javítandó sort inverzben láthatjuk. Ezután újraírhatjuk a kívánt sort. A javítandó sort az utasítás kiadása után megjelenő kérdőjelre kell megadni. Ha nem írunk sorszámot, akkor az előzőleg használt sort veszi. Ha az újraírásnál csak a RETURN-t nyomjuk le, akkor a programsor nem változik.

K: Kezdőcímtől végcímig, kívánt adattal feltölti a memóriát. (Minden adatot decimálisan vár.)

L: Kezdősorszámtól végsorszámig listázza a forrásprogramot. Ha a „K/P" kérdésre „K"-val felelünk, akkor képernyőre; ha „P"-vel, akkor printerre listázza. Ha képernyőre listázunk, akkor a listázás módja: miután a gép kiírja a sort, le kell nyomni egy billentyűt a folytatáshoz. Ha az „S" billentyűt nyomjuk le, akkor visszatérünk parancs üzemmódba.

M: Kiírja a szabad memóriahelyek számát (decimálisan).

R: Újrasorszámozás. RENUMBER) Megadott kezdőcímtől, megadott lépésközzel újrasorszámozza a forrásprogramot, majd meghívja a listázó rutint.

S: Kiírja a szimbólumtáblát. Lassítható: C= billentyűvel, megállítható: CTRL+S, újraindítható: S

LOAD: A megadott nevű programot a memóriába tölti. Ha a „Lemez/Kazetta" kérdésre „L"-lel válaszolunk, akkor lemezről, ha „K"-val, akkor kazettáról tölti be. Ha csak RETURN-t nyomunk, akkor kazettának veszi.

SAVE: A megadott nevű forrásprogramot kimásolja lemezre vagy kazettára. Lásd még: LOAD!

VERIFY: A tárban levő forrásprogramot összehasonlítja a felvett forrásprogrammal. Ha eltérést talál, hibaüzenetet kapunk. Lásd még: LOAD!

U: Kitörli a forrásprogramot.

X: Visszatérés BASIC programozási üzemmódba.

Utasítások beírása

címke=XXX $nnnn ; megjegyzés Pl. C=LDA #$FF;ab

```
10 REM ********************************
20 REM *      C= UJSAG SORSZAM: 047     *
30 REM *      ASSEMBLER                 *
40 REM *      IFJ. GULYAS LASZLO 1986   *
50 REM ********************************
55 :
60 PS=170:DIMT$(PS),R$(PS),SY$(3*PS),SY(3*PS)
70 TRAP1410:GOTO1430
90 IFU$="STA"ORU$="ADC"ORU$="CMP"ORU$="SBC"ORU$="AND"ORU$="ORA"THEN120
100 IFU$="EOR"ORU$="LDA"THEN120
110 H
120 U$=LEFT$(A$,3):IFU$="LDA"THENY=10:X=1
130 IFU$="STA"THENY=8:X=1
140 IFU$="ADC"THENY=6:X=1
150 IFU$="CMP"THENY=12:X=1
160 IFU$="SBC"THENY=14:X=1
170 IFU$="AND"THENY=2:X=1
180 IFU$="ORA"THENY=0:X=1
190 IFU$="EOR"THENX=1:Y=4
200 R$=RIGHT$(A$,2):IFR$=".X"ORR$=".Y"THENY=Y+1
210 IFMID$(A$,5,1)="("ANDMID$(A$,9,1)=","THENK=7:H=2:X=1:GOTO320
220 IFU$="EOR"ANDMID$(A$,5,1)="("ANDMID$(A$,9,1)<>"."THEN360
230 IFMID$(A$,5,1)="("ANDMID$(A$,9,1)<>"."THENK=7:H=2:X=1:GOTO320
240 IFMID$(A$,5,1)="$"ANDLEN(A$)=7THENX=5:K=6:H=2:GOTO320
250 IFMID$(A$,5,1)="$"ANDLEN(A$)>6THENX=X+12:K=6:H=4
260 IFR$=".X"ANDMID$(A$,8,1)<>"."THEN320
270 IFR$=".Y"ANDMID$(A$,5,1)="$"THENX=X-4:GOTO320
280 IFR$=".X"ANDMID$(A$,8,1)="."THENK=6:H=2:X=5:GOTO320
290 IFMID$(A$,5,1)="$"ANDLEN(A$)<=6THENX=5:K=6:H=2
300 IF(R$=".Y"ORR$=".X")ANDMID$(A$,5,1)="."THENK=6:H=2
310 IFMID$(A$,5,1)="#"THENX=9:K=7:H=2
320 PRINTTAB(0)"$"HEX$(CP),TAB(6)CP;:C$="":A=Y*16+X:GOSUB10000:IFR=137THENH
325 IFMID$(A$,5,1)="#"ANDLEN(A$)>8THENH
330 IFH=0THEN340:ELSEA$=MID$(A$,K,H):IFH=2THENA=DEC(A$):GOSUB10000:ELSEGOSUB370
340 PRINTTAB(12)C$TAB(20);D$;TAB(29)AA$
360 RETURN
370 A=DEC(RIGHT$(A$,2)):GOSUB10000:A=DEC(LEFT$(A$,2)):GOSUB10000:RETURN
420 IFA$="BRK"THENX=0:Y=0:H=0:GOTO320
430 IFA$="PHP"THENX=8:Y=0:H=0:GOTO320
440 IFA$="CLC"THENX=8:Y=1:H=0:GOTO320
450 IFA$="SEC"THENX=8:Y=3:H=0:GOTO320
460 IFA$="PLP"THENX=8:Y=2:H=0:GOTO320
470 IFA$="PHA"THENX=8:Y=4:H=0:GOTO320
480 IFA$="CLI"THENX=8:Y=5:H=0:GOTO320
490 IFA$="PLA"THENX=8:Y=6:H=0:GOTO320
500 IFA$="SEI"THENX=8:Y=7:H=0:GOTO320
510 IFA$="RTS"THENX=0:Y=6:H=0:GOTO320
520 IFA$="RTI"THENX=0:Y=4:H=0:GOTO320
530 IFA$="SED"THENX=8:Y=15:H=0:GOTO320
540 IFA$="CLD"THENX=8:Y=13:H=0:GOTO320
550 IFA$="DEY"THENX=8:Y=8:H=0:GOTO320
560 IFA$="TYA"THENX=8:Y=9:H=0:GOTO320
570 IFA$="TAY"THENX=8:Y=10:H=0:GOTO320
580 IFA$="CLV"THENX=8:Y=11:H=0:GOTO320
590 IFA$="INY"THENX=8:Y=12:H=0:GOTO320
600 IFA$="INX"THENX=8:Y=14:H=0:GOTO320
610 IFA$="NOP"THENX=10:Y=14:H=0:GOTO320
620 IFA$="DEX"THENX=10:Y=12:H=0:GOTO320
630 IFA$="TSX"THENX=10:Y=11:H=0:GOTO320
640 IFA$="TAX"THENX=10:Y=10:H=0:GOTO320
650 IFA$="TXS"THENX=10:Y=9:H=0:GOTO320
660 IFA$="TXA"THENX=10:Y=8:H=0:GOTO320
670 U$=LEFT$(A$,3)
680 IFU$<>"ASL"ANDU$<>"ROL"ANDU$<>"LSR"ANDU$<>"ROR"THEN790
690 IFU$="ASL"THENX=6:Y=0
700 IFU$="ROL"THENX=6:Y=2
710 IFU$="LSR"THENX=6:Y=4
720 IFU$="ROR"THENX=6:Y=6
730 IFLEN(A$)=3THENX=X+4:H=0:GOTO320
740 R$=RIGHT$(A$,2):IFLEFT$(R$,1)="."ANDR$<>".X"THENH
750 IFR$=".X"THENY=Y+1
760 IFMID$(A$,5,1)="$"AND(LEN(A$)=7ORMID$(A$,8,1)=".")THENX=6:H=2:K=6:GOTO320
770 IFMID$(A$,5,1)="$"AND(LEN(A$)=9ORMID$(A$,10,1)=".")THENX=14:H=4:K=6:GOTO320
790 U$=LEFT$(A$,3):IFU$<>"JMP"THEN820
800 IFMID$(A$,5,1)="$"THENX=12:Y=4:H=4:K=6:GOTO320
810 IFMID$(A$,5,1)="("THENX=12:Y=6:H=4:K=7:GOTO320
820 IFU$<>"LDX"ANDU$<>"LDY"THEN920
830 IFU$="LDX"THENX=2:Y=10
840 IFU$="LDY"THENX=0:Y=10
850 IFMID$(A$,5,1)="#"THENK=7:H=2:GOTO320
860 R$=RIGHT$(A$,2)
870 IFR$=".X"ANDU$="LDX"THENH
880 IFR$=".Y"ANDU$="LDY"THENH
890 IFR$=".X"ORR$=".Y"THENY=Y+1
900 IFMID$(A$,5,1)="$"AND(LEN(A$)=7ORMID$(A$,8,1)=".")THENX=X+4:H=2:K=6:GOTO320
910 IFMID$(A$,5,1)="$"AND(LEN(A$)=9ORMID$(A$,10,1)=".")THENX=X+12:H=4:K=6:GOTO32
0
920 IFU$<>"STX"ANDU$<>"STY"THEN1010
930 IFU$="STX"THENX=6:Y=8
940 IFU$="STY"THENX=4:Y=8
950 R$=RIGHT$(A$,2)
960 IFR$=".X"ANDU$="STX"THENH
970 IFR$=".Y"ANDU$="STY"THENH
980 IFMID$(A$,5,1)="$"ANDLEN(A$)=7THENK=6:H=2:GOTO320
990 IFMID$(A$,5,1)="$"ANDMID$(A$,8,1)="."THENY=Y+1:K=6:H=2:GOTO320
1000 IFMID$(A$,5,1)="$"ANDMID$(A$,10,1)="."THENH:ELSEK=6:H=4:X=X+8:GOTO320
1010 IFU$<>"INC"ANDU$<>"DEC"THEN1090
1020 IFU$="DEC"THENX=6:Y=12
1030 IFU$="INC"THENX=6:Y=14
1040 R$=RIGHT$(A$,2):IFR$=".Y"THENH
1050 IFR$=".X"THENY=Y+1
1060 IFMID$(A$,5,1)="$"AND(LEN(A$)=7ORMID$(A$,8,1)=".")THENX=6:H=2:K=6:GOTO320
1070 IFMID$(A$,5,1)="$"AND(LEN(A$)=9ORMID$(A$,10,1)=".")THENX=14:H=4:K=6:GOTO320
1090 IFU$<>"CPX"ANDU$<>"CPY"THEN1150
1100 IFU$="CPX"THENX=0:Y=14
1110 IFU$="CPY"THENX=0:Y=12
1111 IFMID$(A$,LEN(A$)-1,1)="."THENH
1120 IFMID$(A$,5,1)="#"THENK=7:H=2:GOTO320
1130 IFMID$(A$,5,1)="$"ANDLEN(A$)=7THENX=X+4:K=6:H=2:GOTO320
1140 IFMID$(A$,5,1)="$"ANDLEN(A$)>7THENX=X+8:K=6:H=4:GOTO320
1150 IFU$<>"BIT"ANDU$<>"JSR"THEN1190
1160 IFU$="JSR"ANDLEN(A$)=9THENX=0:Y=2:K=6:H=4:GOTO320
1170 IFU$="BIT"ANDLEN(A$)=7THENX=4:Y=2:K=6:H=2:GOTO320
1180 IFU$="BIT"ANDLEN(A$)=9THENX=12:Y=2:K=6:H=4:GOTO320
1190 IFU$="BPL"ORU$="BMI"ORU$="BVC"ORU$="BVS"ORU$="BCC"ORU$="BCS"THEN1220
1200 IFU$="BNE"ORU$="BEQ"THEN1220
1210 GOTO90
1220 IFU$="BPL"THENA=16
1230 IFU$="BMI"THENA=48
1240 IFU$="BVC"THENA=80
1250 IFU$="BVS"THENA=112
1260 IFU$="BCC"THENA=144
1270 IFU$="BCS"THENA=176
1280 IFU$="BNE"THENA=208
1290 IFU$="BEQ"THENA=240
1300 PRINTTAB(0)"$"HEX$(CP)TAB(6)CP;
```

ahol: címke – tetszőleges név
XXX – hárombetűs mnemonik
$nnnn – adat

Ha nem írunk mnemonikot a címke után, akkor a program úgy veszi, hogy a címkének akarunk értéket adni. Ennek módja: CIM = $0C00. Az adat csak hexadecimális lehet!

Az adat lehet:
#$ egy byte-os Pl. LDA #$FF
Snnnn két byte-os Pl. STA SFFFF
IIII címke. Ha egy byte-os, akkor #$-ként, ha két byte-os, akkor $nnnn-ként fordít. Pl. LDA IIII
*IIII nullalapos címke. Csak egy byte-os. Pl. LDA *IIII

A címkék használatakor írhatunk hexában plusz, vagy mínusz 255-öt. Pl.
LDA IIII + $0C vagy
LDA IIII + $FF
LDA IIII – $0C vagy
LDA IIII – $FF

A címke tetszőleges hosszúságú lehet, és elhagyható.

Minden használt két byte-os címkének használhatjuk külön az alsó, és külön a felső byte-ját.
IIII = $0CFF
IIII a felső byte-ja #$0C
IIII az alsó byte-ja #$FF

Az egy byte-os adatok helyett, ha pl. 'A'-t írunk be, akkor az aposztrofok között levő karakter ASCII kódját fordítja le a program #$2A módon. Pl. LDA 'A' = LDA #$41

Lehetőségünk van még, a BASIC DATÁ-jához hasonlóan adatokat bevinni megadott címre. A töltési címet a „*" címke tartalmazza. Az adatokat „."-tal (ponttal) választjuk el, a „.BY" utasításon belül. Vihetünk be adatokat hexában (Pl. $FF),, vagy „'"-ok (aposztrofok) közé írt karakterek ASCII kódjaként. (Pl. '6'.)

A programban azokat az assembly utasításokat, amelyekben indexelt címzés szerepel, LDA $FFFF,X helyett, LDA $FFFF.X módon kell beírni. Tehát vessző helyett pontot kell írni, az indexelt címzésnél.

A 64K-s változatban vagy Plusz 4-esen lehetőségünk van a képernyőre szöveget írni. Ezt a „TEXT" utasítással érhetjük el.

Pl. TEXT SZIA!
Ha a „TEXT" utasítás előtt be akarjuk állítani a kurzort, akkor a PLOT rutin meghívása után az Y regiszterbe töltsünk 4-et! (LDY#$04).

Hibaüzenetek

Hiba 1 Nincs bekapcsolva a hívott egység. Pl. printer.
Hiba 2 Nincs több hely a forrásprogram számára.
Hiba 3 A felvett program nem egyezik a tárban levővel.
Hiba 4 Nem definiáltuk a keresett szimbólumot.
Huh!? A program számára értelmetlen szöveg.

A programban szerepel néhány értelmetlennek tűnő utasítás, ezeket kérem változtatás nélkül írják be.

Jótanácsok

Ha object módban kívánjuk a programot kimenteni a TEDMON segítségével, akkor az „E" paranccsal nézzük meg a címpárt a kimentéshez. Ha van a programban TEXT utasítás, akkor a végcím nem fog egyezni a fordításnál kiírttal, mert a szöveg adatait a program után tölti az ASSEMBLER, és az „E" parancsnál ezeknek a végét írja ki.

Vigyázat! Az ASSEMBLER LOAD utasítás nem törli ki az előző forrásprogramot.

Ha valamilyen okból a programból BASIC-be tértünk vissza, és úgy szeretnénk visszatérni, hogy forrásprogramunk megmaradjon, akkor gépeljük be a GOTO70 utasítást.

A 16K-s változatot úgy készíthetjük el, hogy:
a 60-as sorban a PS változót 170-ről 50-re javítjuk,
a 1895-ös, és a 2505-ös sort töröljük,
és kiadjuk a DELETE3165–3270 utasítást.

```
1301 C$="":GOSUB10000:A=DEC(MID$(A$,6,4)):IFA<PO-128ORA>PO+127THENH
1310 A=255-(PO-A):IFA>255THENA=A-256
1320 GOSUB10000:GOTO340
1410 CLOSE1:IFER=5THENIH=1:GOTO10001
1420 TRAP1410:PRINT" HUH!? "I:GOTO1660
1430 PRINT"■ASSEMBLER■■":IFIH=1THENPRINT"HIBA 1"
1470 I=0:KC=0:VC=0:IH=0
1480 INPUT"*";P$
1490 IFP$="I"THEN1670
1500 IFP$="A"THEN2970
1510 IFP$="F"THEN1790
1520 IFP$="J"THEN1970
1530 IFP$="L"THEN2010
1540 IFP$="VERIFY"THEN2320
1550 IFP$="G"THEN2130
1560 IFP$="U"THENRUN
1570 IFP$="D"THEN2160
1580 IFP$="K"THEN2190
1590 IFP$="X"THENEND
1600 IFP$="M"THENPRINTFRE(0)
1610 IFP$="R"THEN2230
1620 IFP$="S"THENGOTO3000
1630 IFP$="E"THENPRINTPP,PX
1640 IFP$="LOAD"THEN2320
1650 IFP$="SAVE"THEN2320
1660 GOTO1470
1670 INPUT"AUTO";V$
1680 IFV$="I"THEN1720
1690 PRINT,:INPUTP,T$(P):AK=P:IFP>VPTHENVP=P
1700 IFT$(P)="."THENT$(P)="":GOTO1660
1710 GOTO1690
1720 INPUT"K.,LE";K,L
1730 IFK>PSTHEN1770
1740 A$=T$(K)
1750 PRINTK,:INPUTT$(K):AK=K:IFK>VPTHENVP=K
1760 IFT$(K)="."THENT$(K)=A$:GOTO1660
1770 K=K+L:IFK>PSTHENPRINT"HIBA 2":GOTO1660:ELSEGOTO1740
1780 IFW$="'"THENER=ASC(MID$(A$,6,1)):A$=LEFT$(A$,4)+"#$"+RIGHT$(HEX$(ER),2)
1790 INPUT"F.C.";PO:PP=CP:INPUT"T.C.";CP
1800 CHAR1,2,24,"■■":SY=0:GOTO2440
1820 CHAR1,2,24," ":PRINT:FORI=0TOPS
1830 IFT$(I)=""THENNEXTI:GOTO1660
1840 D$="":A=INSTR(T$(I),";",1):IFA=0THENA$=T$(I):GOTO1860
1850 IFA=1THENA$=T$(I):NEXTI:GOTO1660:ELSEA$=LEFT$(T$(I),A-1)
1860 Z=INSTR(A$,"=",1):IFZ=0ORZ=1THENA$=A$:GOTO1870:ELSEA$=RIGHT$(A$,LEN(A$)-Z)
1870 IFLEFT$(A$,1)="$"THENNEXTI:GOTO1660
1880 GOTO3120
1890 IFMID$(A$,5,1)="'"THENA$=LEFT$(A$,4)+"#$"+RIGHT$(HEX$(ASC(MID$(A$,6,1))),2)
1895 IFLEFT$(A$,5)="TEXT "THEN3170
1896 AA$=A$:W$=MID$(A$,5,1)
1900 IFW$<>"#"ANDW$<>"$"ANDLEN(A$)>3ANDMID$(A$,6,1)<>"$"ANDW$<>"("THENQ=5:GOTO027
30
1910 IFW$="*"THENQ=5:GOTO2730
1920 IF(W$="#"ANDMID$(A$,6,1)<>"$")OR(W$="("ANDMID$(A$,6,1)<>"$")THENQ=6:GOTO273
0
1930 IFZ<>0ANDZ<>1THEND$=LEFT$(T$(I),Z)
1940 GOSUB420:NEXTI:GOTO1660
1970 J=AK:INPUTJ:IFJ<10THENKC=0:ELSEKC=J-10
1980 IFJ>PS-10THENVC=PS:ELSEVC=J+10
1990 FORI=KCTOVC:IFI=JTHENPRINT"■";I,T$(I);"■":ELSEIFT$(I)<>""THENPRINTI,T$(I)
2000 NEXTI:INPUTT$(J):GOTO1660
2010 INPUT"K.C,V.C";KC,VC
2020 PRINT"■K■/■N■"
2030 GETA$
2040 IFA$<>"K"ANDA$<>"N"THEN2030
2050 IFA$="N"THENGOSUB3140:GOTO1660
2060 FORI=KCTOVC
2070 IFT$(I)=""THENNEXTI:GOTO1660:ELSEPRINTI,T$(I):AK=I
2090 GETA$:IFA$=""THEN2090
2100 IFA$="S"THEN1660:ELSENEXTI:GOTO1660
2130 INPUT"S.C.";G:SYSG:GOTO1660
2160 INPUT"K.C.,V.C.";KC,VC:FORI=KCTOVC:T$(I)="":NEXTI:KC=0:VC=PS:GOTO2060
2190 INPUT"K.C,.C,BY";KC,VC,BY:FORI=KCTOVC:POKEI,BY:NEXTI:GOTO1660
2230 FORI=0TOPS:R$(I)="":NEXTI:INPUT"K.C,LEP";KC,LE:FORI=0TOPS
2240 IFT$(I)=""THENNEXTI:GOTO2290
2270 R$(KC)=T$(I):KC=KC+LE:IFKC>PSTHENPRINT"HIBA 2":GOTO1660:ELSENEXTI
2290 FORI=0TOPS:T$(I)="":T$(I)=R$(I):NEXTI:KC=0:VC=PS:GOTO2060
2320 INPUT"NEV";N$:INPUT"■L■EMEZ/■K■AZETTA";V$:IFV$="L"THENES=8:ELSEES=1
2330 IFP$="LOAD"THEN2390:ELSEIFP$="SAVE"THEN2380
2350 OPEN1,ES,0,N$:INPUT#1,VP:FORI=1TOVP:INPUT#1,V$:IFV$="π"THENCLOSE1:GOTO1660
2360 IFV$<>T$(I)THENPRINT:PRINT"HIBA 3":CLOSE1:GOTO1660:ELSENEXTI:CLOSE1:GOTO166
0
2380 OPEN1,ES,1,N$:PRINT#1,VP:FORI=1TOVP:PRINT#1,T$(I):NEXTI:PRINT#1,"π":CLOSE1:
GOTO1660
2390 OPEN1,ES,0,N$:INPUT#1,VP:FORI=1TOVP:INPUT#1,T$(I):IFT$(I)="π"THENT$(I)="":I
=VP+1
2400 NEXTI:CLOSE1:GOTO1660
2440 P1=PO:FORI=0TOPS:IFT$(I)=""THENNEXTI:GOTO2476
2460 A=INSTR(T$(I),";",1):IFA=0THENA$=T$(I):GOTO2475
2470 IFA=1THENA$="":GOTO2620:ELSEA$=LEFT$(T$(I),A-1)
2475 A=INSTR(A$,"=",1):IFA<>0ANDA<>1ANDMID$(A$,A+1,1)="$"THENGOSUB2670:NEXTI:ELS
ENEXTI
2476 FORI=0TOPS:IFT$(I)=""THEN2620:ELSEA=INSTR(T$(I),";",1):IFA=0THENA$=T$(I):GO
TO2480
2478 IFA=1THENA$="":GOTO2620:ELSEA$=LEFT$(T$(I),A-1)
2480 A=INSTR(A$,"=",1):IFA=0THENA$=A$:GOTO2500
2490 IFA=1THENA$=A$:ELSEIFMID$(A$,A+1,1)<>"$"THEN2690
2495 IFA>1THEN2620
2500 U$=LEFT$(A$,3):IFLEFT$(A$,1)="$"THEN2620
2505 IFLEFT$(A$,5)="TEXT "THENP1=P1+39
2506 IFLEFT$(A$,4)=".BY "THENP1=P1-2
2510 IFU$="PHP"ORU$="CLC"ORU$="PLP"ORU$="SEC"ORU$="PHA"ORU$="CLI"ORU$="PLA"THEN2
700
2520 IFU$="SEI"ORU$="DEY"ORU$="TYA"ORU$="TAY"ORU$="CLV"ORU$="INY"ORU$="CLD"THEN2
700
2530 IFU$="INX"ORU$="SED"ORU$="BRK"ORU$="RTI"ORU$="RTS"ORU$="NOP"ORU$="DEX"THEN2
700
2540 IFU$="TSX"ORU$="TAX"ORU$="TXS"ORU$="TXA"THEN2700
2550 IFU$="JSR"ORU$="JMP"THENP1=P1+3:GOTO2620
2560 IFU$="BPL"ORU$="BMI"ORU$="BVC"ORU$="BVS"ORU$="BCC"ORU$="BCS"ORU$="BNE"THEN2
710
2570 IFU$="BEQ"THEN2710
2580 IFMID$(A$,5,1)="#"ORMID$(A$,5,1)="<"ORMID$(A$,5,1)=">"ORMID$(A$,5,1)="'"THE
N2710
2590 IFMID$(A$,5,1)="$"AND(MID$(A$,8,1)="."ORLEN(A$)=7)THEN2710
2595 IFMID$(A$,5,1)="*"ORMID$(A$,6,1)="*"THENQ=6:GOTO2710:ELSEIFMID$(A$,5,1)="("
THEN2610
2601 A=INSTR(A$,"+",5):IFA=0THEN2602:ELSE2605
2602 A=INSTR(A$,"-",5):IFA=0THEN2603:ELSE2605
2603 A=INSTR(A$,".",5):IFA=0THEN2604:ELSE2605
2604 A=LEN(A$)+2
2605 S1=LEN(A$)-A+1:SY$=MID$(A$,5,LEN(A$)-S1-5)
2606 FORW=0TOSY:IFSY$=SY$(W)ANDSY(W)<256THENW=SY+1:GOTO2710:ELSENEXTW
2610 P1=P1+3
2620 NEXTI:PX=CP+(P1-PO)
2630 FORI=0TOSY-1:SY$(SY)="<"+SY$(I):SY(SY)=DEC(RIGHT$(HEX$(SY(I)),2))
2640 IFSY$(I)="*"THENMU=SY(I)
2650 SY=SY+1:SY$(SY)=">"+SY$(I):SY(SY)=DEC(LEFT$(HEX$(SY(I)),2)):SY=SY+1:NEXTI
2660 GOTO1820
2670 SY$(SY)=LEFT$(A$,A-1):A$=RIGHT$(A$,LEN(A$)-A)
2680 SY(SY)=DEC(MID$(A$,2,4)):SY=SY+1:RETURN
2690 SY$(SY)=LEFT$(A$,A-1):A$=RIGHT$(A$,LEN(A$)-A):SY(SY)=P1:SY=SY+1:GOTO2500
2700 P1=P1+1:GOTO2620
2710 P1=P1+2:GOTO2620
2730 IFMID$(A$,5,2)="(*"ANDRIGHT$(A$,1)="Y"THENQ=6+1:W=0:W1=1:ELSEW=1:W1=0
2740 IFMID$(A$,5,2)="(*"ANDRIGHT$(A$,2)<>".Y"THENQ=6+1:W=1:QW=1:W1=1:ELSEQW=0:W1
=0
```

```
2750 A=INSTR(A$,"+",Q):IFA=0THEN2770:ELSEQ$="+":GOTO2840
2770 A=INSTR(A$,"-",Q):IFA=0THEN2790:ELSEQ$="-":GOTO2840
2790 A=INSTR(A$,".",Q):IFA=0THEN2810:ELSEQ$="":GOTO2840
2810 A=INSTR(A$,")",Q):IFA=0THEN2830:ELSEQ$="":W=1:GOTO2840
2830 Q$="":A=LEN(A$)+1
2840 S1=LEN(A$)-A+1:SY$=MID$(A$,O,LEN(A$)-S1-Q+W)
2850 IFW$="*"THENSY$=RIGHT$(SY$,LEN(SY$)-1)
2860 FORR=0TOSY-1:IFSY$=SY$(R)THEN2890:ELSENEXTR
2880 PRINT"HIBA 4":GOTO1660
2890 EK=SY(R):IFQ$="+"ORQ$="-"THENE=DEC(MID$(A$,A+2,2)):A=A+4
2900 IFQ$="+"THENEK=EK+E
2910 IFQ$="-"THENEK=EK-E
2920 E$=HEX$(EK):IFEK<256THENE$=RIGHT$(E$,2)
2930 IFW$<>"*"ANDMID$(A$,6,1)<>"*"ANDWi<>1ANDLEN(E$)=2THENHX$="#$"+E$:ELSEHX$="$
"+E$
2940 B$=LEFT$(A$,Q-1-(1-W)-QW)+HX$
2950 IFA<=LEN(A$)THENB$=B$+RIGHT$(A$,LEN(A$)-A+1+(1-W))
2960 A$=B$:Q=0:GOTO1930
2970 INPUTS$:IFLEFT$(S$,1)="$"THENPRINTDEC(MID$(S$,2,4)):GOTO1660
2990 PRINT"$"+HEX$(VAL(S$)):GOTO1660
3000 FORI=0TOSY-1:PRINTSY$(I),:IFSY(I)<256THENPRINT"#";
3010 PRINT"$";:IFSY(I)<256THENH$=RIGHT$(HEX$(SY(I)),2):ELSEH$=HEX$(SY(I))
3020 PRINTH$:NEXTI
3030 GOTO1470
3050 A=INSTR(A$,".",PW)
3060 W$=MID$(A$,PW,1)
3070 IFW$="'"THENEK=ASC(MID$(A$,PW+1,1))
3080 IFW$="$"THENEK=DEC(MID$(A$,PW+1,2))
3090 POKEMU,EK:MU=MU+1:IFA<>0ANDA<>LEN(A$)THENPW=A+1:ELSE3110
3100 IFPW=LEN(A$)-3THENPW=LEN(A$):GOTO3060:ELSEGOTO3050
3110 NEXTI:GOTO1660
3120 IFLEFT$(A$,4)=".BY "THENPW=5:GOTO3050:ELSE1890
3140 OPEN1,4:CMD1:FORI=KCTOVC:IFT$(I)=""THEN3160:ELSEPRINT#1,I,T$(I)
3160 NEXTI:CLOSE1:RETURN
3170 IFZ<>0ANDZ<>1THEND$=LEFT$(T$(I),Z)
3171 PRINTTAB(0)"$"HEX$(CP);TAB(6)CP;TAB(20);D$;TAB(29);"TEXT"
3175 RESTORE3260:FORR=0TO41:READA:POKECP,A:CP=CP+1:PO=PO+1:NEXTR
3180 C$=HEX$(PX):E=DEC(RIGHT$(C$,2)):M=DEC(LEFT$(C$,2)):POKECP-11,E:POKECP-10,M
3190 POKECP-14,E:POKECP-13,M:POKECP-23,E:POKECP-22,M:POKECP-34,E:POKECP-33,M
3200 PX=PX+1
3210 C$=HEX$(PX-5):E=DEC(RIGHT$(C$,2)):M=DEC(LEFT$(C$,2))
3220 POKECP-20,E:POKECP-19,M:POKECP-8,LEN(A$)
3240 FORR=6TOLEN(A$):POKEPX,ASC(MID$(A$,R,1)):PX=PX+1:NEXTR
3250 NEXTI:GOTO1660
3260 DATA72,138,72,152,72,162,0,141,0,101,56,32,240,255,24,32,240,255,174,0,101,
189
3270 DATA0,128,32,210,255,238,0,101,174,0,101,224,35,208,229,104,168,104,170,104
,96
10000 POKECP,A:PO=PO+1:CP=CP+1:C$=C$+RIGHT$(HEX$(A),2):RETURN
10001 POKE1319,71:POKE1320,207:POKE1321,55:POKE1322,48:POKE1323,13:POKE239,5:POK
E2035,1
10002 H
28200 Q$="":IFA=LEN(A$)ANDRIGHT$(A$,3)<>".X)"THENW=1:GOTO2840:ELSEW=0:GOTO2840
```

C= 16

C= PLUS 4

MONITOR MÉG EGYSZER

5. számunkban sajnos hibásan jelent meg a VC20-asra írt monitor program listája. A lista bizonyos részeit felcseréltük, de ez nyilván egyetlen olvasónknak sem okozott gondot. Annál inkább az olvashatatlan rész, amit most újra közlünk.

```
8150 REM**VECTORC,B**
8160 REM-24433-
8162 DATA32,106,90,32,221,94,160,0,165,20,120,145,174,200,165,21,145,174,88,96
8200 REM**REGISTERS**
8210 REM-24453-
8211 REM-AXYFSP-
8212 DATA65,88,89,70,83,80
8214 REM-24459-
8215 DATA32,7,90,32,1,90,173,17,3,133,183,173,18,3,133,184,32,223,90
8220 DATA162,251,134,10,32,31,90,32,7,90,166,10,189,138,94,32,210,255,32,28,90,1
89,17,2
8225 DATA32,249,90,230,10,208,229,76,31,90
8227 REM*,REG*NUM****
8228 REM-24512-
8230 DATA72,32,115,0,104,201,80,208,10,32,106,90,142,18,3,140,17,3,96,160,251,21
7
8232 DATA138,94,208,12,132,10,32,158,215,164,10,138,153,17,2,96,200,208,236,76,8
,207
8235 REM**SZOVEG*****
8237 REM-24555-
8240 DATA141,32,86,73
8241 DATA67,77,79,78,32
8242 DATA66,89,32,71,46
8243 DATA32,84,79
8244 DATA84,72,141
8999 DATA-1
9000 REM**************
9001 REM***ELEJE*****
9002 REM**************
9010 REM
9020 REM***EDITOR****
9030 REM-22529-
9032 DATA8,174,52,3,172,53,3,134,183,132,184,162,253,40,76,21,88,32,115,0,240,36
9034 DATA157,57,2,232,208,245,160,0,138,24,105,3,133,10,189,139,91,217,54,3,208,
8
9036 DATA232,200,228,10,208,242,240,9,166,10,224,171,144,226,76,8,207,134,187
9040 REM***2.LAP******
9041 DATA160,0,32,115,0,240,12,201,65,208,22,32,115,0,208,235,169,2,44,169,0,133
,188
9043 DATA32,212,88,176,223,76,250,86,234,234,201,35,240,4,201,40,208,7,153,54,3,
200,32
9045 DATA115,0,169,0,153,54,3,200,132,10,32,63,89,160,32,138,208,2,160,5,132,99,
164,10
9046 REM***3.LAP*****
9047 DATA32,121,0,76,143,88,32,115,0,240,10,153,54,3,200,192,7,144,243,176,160,1
69,0
9050 DATA153,54,3,32,10,89,32,212,88,144,8,165,99,201,46,208,242,240,139
9060 DATA234,234,76,222,86,165,20,145,183,165,188,201,32,144,7,32,117,90
9070 DATA165,21,145,183,32,117,90,166,183,164,184,142,52,3,140,53,3,96
9100 REM**SZOKERESES*
9110 REM-22740-
9111 DATA162,93,160,0,134,14,132,13,132,10,160,37,177,13,240,26
9112 DATA197,187,208,17,200,208,2,230,14,177,13,197,188,208,11,24,165,10,160,0
9113 DATA96,200,208,2,230,14,200,208,2,230,14,230,10,208,217,56,176,236
9200 REM*CIMKERESES**
9210 REM-22794-
9220 DATA166,99,134,188,160,2,132,10,160,0,189,51,92,208,2,198,10,217,54,3,208,9
,232,200
9221 DATA165,10,208,238,134,99,96,165,10,240,10,232,189,51,92,208,250,198,10,208
,246
9225 DATA232,224,46,144,208,76,8,207
9227 REM*SZBEO*REL***
9229 REM-22847-
9240 DATA76,179,86
9245 DATA165,187,201,15,144,45,201,39,176,41,32,27,86,165,20,24,229,183,168,165,
21
9246 DATA229,184,170,240,8,232,208,23,152,16,20,48,5,208,16,152,48,13,134,21,132
,20
9248 DATA165,183,208,2,198,184,198,183,96,76,72,210
9300 REM***START*****
9305 REM-22905-
9306 DATA162,236,189,0,95,32,210,255,232,208,247,162,1,160,84,142,10,3,140,11,3
9307 DATA162,153,160,89,142,8,3,140,9,3,96
9310 REM***NEWCODE1**
9312 REM-22937-
9315 DATA32,115,0,201,91,240,6,32,121,0,76,231,199,162,124,160,86,142,4,3,140,5,
3
9316 DATA162,217,160,89,32,146,89,76,217,89
9319 REM****STOP*****
9320 REM-22970-
9322 DATA32,225,255,208,217,165,122,164,123,166,58,232,240,12,133,61,132,62,165,
57
```

VC= 20

C= 64

ÓRA

Sokszor jól jöhet egy, a képernyőn mindig látható óra, amely a gépben zajló egyéb dolgoktól függetlenül jár. Íme egy ilyen program, amelyről el kell árulni, hogy bizonyos programokra rossz hatással volt.

A program futtatása (RUN) után a gép először megkérdezi azt a dátumot és időt, amikor indítani akarjuk, majd billentyűnyomásra indul. Az órát a SYS 49711 paranccsal lehet kikapcsolni, ill. bekapcsolni, s a kettő között az óra tovább jár. A SYS 49152, „óra", „perc", „sec", „nap", hányadika, hónap, év (pl.: 86) paranccsal lehet az órát átállítani, majd az említett módon be kell kapcsolni.

```
100 REM ********************************
105 REM *   C= UJSAG  SORSZAM 045     *
110 REM *   ORA                        *
115 REM *   DATA WELWT 85/12 68.OLD.  *
120 REM *   PROGRAM: GERHARDT HAAS     *
125 REM ********************************
130 FORI=49152TO49879
140 READA
150 POKEI,A
160 S=S+A:D=A-D
170 NEXT
180 IFS<>84055ORD<>427THENPRINT"HIBA A D
ATA SOROKBAN !!!":END
190 POKE53280,12:POKE53281,12
200 INPUT"██A PONTOS IDO (OOPPMM) ";A$
210 INPUT"DATUM (EEHHNN) ";B$
220 O$=LEFT$(A$,2):P$=MID$(A$,3,2):M$=RI
GHT$(A$,2)
230 N=VAL(RIGHT$(B$,2)):H=VAL(MID$(B$,3,
2)):E=VAL(LEFT$(B$,2))
240 INPUT"MILYEN NAP VAN ";C$
250 PRINT"█      "O$"."P$"."M$" "C$
255 PRINT"███████████████████████████"RIG
HT$(B$,2)"."MID$(B$,3,2)".19"LEFT$(B$,2)
260 PRINT"████████NYOMJ LE EGY GOMBOT ES
AZ ORA INDUL !!!"
270 POKE198,0:WAIT198,1
280 SYS49152,O$,P$,M$,C$,N,H,E
290 PRINT"█";:POKE198,0:END
32000 DATA32,3,194,201,2,208,100,173,14,
220,9,128,141,14,220,160,0,32,41,194
32001 DATA201,3,176,83,32,34,194,201,10,
176,76,5,2,201,37,176,70,201,19,144,7
32002 DATA248,56,233,18,216,9,128,141,11
,220,32,116,192,141,10,220,32,116,192
32003 DATA141,9,220,169,0,141,8,220,32,4
7,194,32,3,194,160,0,162,255,232,224
32004 DATA7,208,5,162,6,76,142,192,177,3
4,221,66,194,208,239,200,177,34,221,73
32005 DATA194,240,39,136,76,78,192,32,47
,194,32,72,178,76,128,164,32,3,194,201
32006 DATA2,208,240,160,0,32,41,194,201,
6,176,231,32,34,194,5,2,201,96,176,222
32007 DATA96,76,208,194,32,193,193,142,1
65,194,32,193,193,224,13,176,205,142
32008 DATA166,194,32,193,193,224,99,176,
195,142,167,194,138,41,3,208,3,169,29
32009 DATA44,169,28,141,81,194,174,166,1
94,202,189,80,194,205,165,194,144,167
32010 DATA88,96,165,214,208,5,230,214,32
,108,229,206,162,194,240,3,76,49,234
32011 DATA160,0,140,163,194,169,6,141,16
2,194,32,202,193,173,11,220,72,201,18
32012 DATA208,71,173,10,220,208,66,173,9
,220,208,61,173,8,220,208,56,173,164
32013 DATA194,201,6,144,5,169,255,141,16
4,194,238,164,194,238,165,194,174,166
32014 DATA194,202,189,80,194,24,105,1,20
5,165,194,208,8,238,166,194,169,1,141
32015 DATA165,194,173,166,194,201,13,208
,8,169,1,141,166,194,238,167,194,104
32016 DATA201,146,240,18,16,16,41,127,20
1,18,144,5,169,0,76,74,193,24,248,105
32017 DATA18,216,72,216,41,48,74,74,74,7
4,24,105,48,32,241,193,104,41,15,24,105
32018 DATA48,32,241,193,32,239,193,173,1
0,220,32,12,194,32,239,193,173,9,220
32019 DATA32,12,194,173,8,220,32,199,193
,173,164,194,133,2,10,10,24,101,2,10
32020 DATA170,189,92,194,48,7,32,241,193
,232,76,133,193,41,127,32,241,193,32
32021 DATA199,193,173,165,194,32,216,193
,32,239,193,173,166,194,32,216,193,32
32022 DATA239,193,169,49,32,241,193,169,
57,32,241,193,173,167,194,32,216,193
32023 DATA32,202,193,76,49,234,32,253,17
4,76,158,183,162,1,44,162,5,134,2,169
32024 DATA32,32,241,193,198,2,208,247,96
,162,255,232,133,2,56,233,10,176,248
32025 DATA138,24,105,48,32,241,193,165,2
,24,105,48,44,169,46,41,191,172,163,194
32026 DATA153,0,4,200,140,163,194,169,0,
153,255,215,96,32,253,174,32,158,173
32027 DATA76,163,182,72,41,240,74,74,74,
74,24,105,48,32,241,193,104,41,15,24
32028 DATA105,48,76,241,193,10,10,10,10,
133,2,200,177,34,56,233,48,96,173,20
32029 DATA3,73,247,120,141,20,3,173,21,3
,73,42,141,21,3,88,96,72,75,83,67,80
32030 DATA83,86,69,69,90,83,69,90,65,31,
28,31,30,31,30,31,31,30,31,30,31,72,69
32031 DATA84,70,79,32,32,32,32,160,75,69
,68,68,32,32,32,32,32,160,83,90,69,82
32032 DATA68,65,32,32,32,160,67,83,85,84
,79,82,84,79,75,160,80,69,78,84,69,75
32033 DATA32,32,32,160,83,90,79,77,66,65
,84,32,32,160,86,65,83,65,82,78,65,80
32034 DATA32,160,2,40,0,1,1,87,0,0,0,0,0
,0,0,0,200,177,34,201,69,208,5,162,2
32035 DATA76,190,194,162,5,169,142,141,1
42,192,169,164,141,143,192,169,194,141
32036 DATA144,192,76,142,192,224,2,208,2
34,240,218,0,0
```

KALAND-JÁTÉK UTOLJÁRA

A Gengszter kalandjáték hibátlan megoldását egyetlen olvasónk küldte el: Glatt Sándor, Budapestről. Voltak más megoldók is, nekik azonban nem sikerült rájönni a játékban lévő – valóban kicsit nehézre sikerült – számrejtvényre. **Kovács László** például végül is megunva a próbálkozást, inkább beírta a programba, hogy **2305 PRINT S.** Ez is megoldás, de kicsit csalafinta.

Nem gondoltuk, hogy ez ilyen nehéz, ezért még a múlt hónapban közzétett megoldó kulcsból is kihagytuk. Nos, ha mondjuk 5, 17, 53-as számokat kapjuk, akkor a megoldás 161. Ha az első számot, 5-öt megszorzom 3-mal és hozzáadok 2-t, akkor megkapom a 2. számot. Ha a 2. számot (17) megint megszorzom 3-mal és hozzáadok 2-t, megkapom a 3. számot (53) stb. A 3 és a 2 nem állandók, amint a gép által a játékban kiírt számok sem minden esetben az 5, 17, 53. De a törvényszerűség mindig ez: 2. szám = 1. szám*A + B; 3. szám = 2.szám*A + B...stb.

A nyertesnek postán elküldtük a Commodore trikót.

Az egymilliomodik COMMODORE 64

A COMMODORE cég már múlt év novemberében eladta az egymilliomodik C 64-est. A nyugatnémet személyi számítógép piacon a COMMODORE részesedése mintegy 72%. A professzionális igényeket is kielégítő AMIGA-ból október végéig 15 000 készüléket értékesített. 1987-ben további két új termékkel kívánnak megjelenni az AMIGA családjában. 1986/87-ben a COMMODORE egymilliárd nyugatnémet márkás forgalomra számít. Az eső árak ellenére az 1985/86-os forgalom 958,9 millió DM-ra emelkedett. (Az előző évi forgalom 926,2 millió DM volt.) Ez megfelel egy 3,5%-os növekedésnek. Az osztrák leányvállalat az előző évi forgalomból 31,9 millió DM-t könyvelhetett el magának. A sikert annak köszönhette, hogy 1986 elején egy önálló társaságot alapított, amely rugalmasabb mozgást tett lehetővé a piacon. A nyugatnémetek évi nyeresége 9,9 millió DM lett. (Korábban ez csak 6,1 millió DM volt.)
Az anyavállalat 889,3 millió dolláros forgalmat bonyolított le és közben 127,9 millió dollárt veszített.

Computer Persönlich
87. 2. szám január 7.

MEMÓRIATÉRKÉP

5. rész

CÍM HEXADECIMÁLIS	CÍM DECIMÁLIS	LEÍRÁS
116F	4463	Nyomkövető (TRACE) üzemmód jelző
1170–1176	4464–4470	Átmeneti tárolás újraszámozáshoz (RENUMBER)
1177–1179	4471–4473	Átmeneti tárolás grafikához
117A–117B	4474–4475	Mutató lebegőpontos/egész konverziós rutinra ($ 849F)
117C–117D	4476–4477	Mutató egész/lebegőpontos konverziós rutinra ($ 793C)
117E–11D5	4478–4565	Sprite sebesség/irány tábla (88 Byte)
11D6–11EA	4566–4586	VIC regiszterek másolása chip frissítésre (21 Byte)
11EB	4587	Mutató felső/alsó karakterkészletre CHAR-hoz
11EC	4588	Mutató felső/grafikus karakterkészletre CHAR-hoz
11ED–11FF	4589–4607	Átmeneti tárolás RECORD parancshoz
1200–1201	4608–4609	Előző BASIC sorszám
1202–1203	4610–4611	Mutató: BASIC utasítás CONTINUE-hoz
1204	4612	PRINT USING $ szimbólum
1205	4613	PRINT USING kitöltő (space) szimbólum
1206	4614	PRINT USING vessző szimbólum
1207	4615	PRINT USING tizedes pont szimbólum
1208	4616	Utolsó hibaszám (TRAP-hoz)
1209–120A	4617–4618	Utolsó hiba sorszáma ($FFFF – nincs hiba)
120B–120C	4619–4620	ON ERROR GOTO sorszám
120D	4621	Segédmutató TRAP-hoz
120E–120F	4622–4623	Mutató a hibaüzenet szövegére
1210–1211	4624–4625	Szövegvég mutató
1212–1213	4626–4627	BASIC által elérhető legmagasabb cím a 0-ás bankon
1214–1215	4628–4629	DO–LOOP-hoz használt terület
1216–1217	4630–4631	Segédtároló a sorszámokhoz
1218	4632	USR ugrás
1219–121A	4633–4634	USR cím (Lo/Hi)
121B–121F	4635–4639	Bemeneti érték RND-hez
1220	4640	Körszegmens foka
1221	4641	Hideg/meleg RESET státusz mutató
1222	4642	PLAY, tempo (sebesség)
1223–1228	4643–4648	PLAY, voice (hangok)
1229–122A	4649–4650	PLAY, ntime (n idő)
122B	4651	PLAY, octave (oktáv)
122C	4652	PLAY, sharp (hegyesség)
122D–122E	4653–4654	PLAY, pitsch (hangmagasság)
122F	4655	PLAY, voice (hang)
1230–1232	4656–4658	PLAY, waveform (hullámforma)
1233	4659	PLAY, dnote (hangjegy)
1234–1238	4660–4664	PLAY, filter adatok
1239–123D	4665–4669	PLAY, envelope adatok
123E	4670	PLAY, parent
123F–1248	4671–4680	PLAY, attack tábla
1249–1252	4681–4690	PLAY, sustain tábla
1253–125C	4691–4700	PLAY, waveform tábla
125D–1266	4701–4710	PLAY, pulzus szélesség alsó
1267–1270	4711–4720	PLAY, pulzus szélesség felső
1271–1275	4721–4725	PLAY, szűrők
1276–1280	4726–4736	PLAY, megszakítási adatok
1281	4737	SOUND, hangtároló
1282–1284	4738–4740	SOUND, idő alsó érték
1285–1287	4741–4743	SOUND, idő felső érték
1288–128A	4744–4746	SOUND, max. alsó érték
128B–128D	4747–4749	SOUND, max. felső érték
128E–1290	4750–4752	SOUND, min. alsó érték
1291–1293	4753–4755	SOUND, min. felső érték
1294–1296	4756–4758	SOUND, direction (irány)
1297–1299	4759–4761	SOUND, lépésszám alsó érték
129A–129C	4762–4764	SOUND, lépésszám felső érték
129D–129F	4765–4767	SOUND, frekvencia alsó érték
12A0–12A2	4768–4770	SOUND, frekvencia felső érték
12A3–12B0	4771–4784	SOUND, átmeneti tárolók
12B1–12B2	4785–4786	POT, átmeneti tárolás
12B3–12B6	4787–4790	WINDOW, átmeneti tárolás
12B7–12FB	4791–4859	SPRDEF és SAVSPR által használt terület
12FC–12FF	4860–4863	BASIC IRQ által használt terület
1300–17FF	4864–6143	Nem használt RAM terület
1800–1BFF	6144–7167	Foglalt függvénybillentyű szoftver alkalmazására
1C00–FEFF	7168–65279	BASIC szöveg (kisfelbontás)
2000–3FFF	8192–16383	Nagyfelb. képernyő (BASIC ekkor $4000-nél kezdődik)
4000–CFFF	16384–53247	BASIC ROM
4000–4002	16384–16386	ROM hideg belépési pont
4003–4005	16387–16389	ROM meleg belépési pont
4006–4008	16390–16392	ROM IRQ belépési pont
AF00–AF02	44800–44802:	JMP $84B4 konverzió: lebegőpontosról egészre
AF03–AF05	44803–44805:	JMP $79C3 konverzió: egészről lebegőpontosra
AF06–AF08	44806–44808:	JMP $8E42 konverzió: lebegőpontosról ASCII kódra
AF09–AF0B	44809–44811:	JMP $8052 konverzió: ASCII kódról lebegőpontosra
AF0C–AF0E	44812–44814:	JMP $8815 konverzió: lebegőpontosról címre
AF0F–AF11	44815–44817:	JMP $8C75 konverzió: címet lebegőpontosra
AF12–AF14	44818–44820:	JMP $882E MEM–FACC (FACC = 1. lebegőpontos akkumulátor)
AF15–AF17	44821–44823:	JMP $8831 ARG–FACC (ARG = 2. lebegőpontos akkumulátor)
AF18–AF1A	44824–44826:	JMP $8845 MEM + FACC
AF1B–AF1D	44827–44829:	JMP $8848 ARG + FACC
AF1E–AF20	44830–44832:	JMP $8A24 MEM*FACC
AF21–AF23	44833–44835:	JMP $8A27 ARG*FACC
AF24–AF26	44836–44838:	JMP $8B49 MEM/FACC
AF27–AF29	44839–44841:	JMP $8B4C ARG/FACC
AF2A–AF2C	44842–44844:	JMP $89CA FACC természetes logaritmusa
AF2D–AF2F	44845–44847:	JMP $8CFB FACC egész része
AF30–AF32	44848–44850:	JMP $8FB7 FACC négyzetgyöke
AF33–AF35	44851–44853:	JMP $8FFA FACC negatívja
AF36–AF38	44854–44856:	JMP $8FBE ARG-ot MEM-edik hatványra emeli
AF39–AF3B	44857–44859:	JMP $8FC1 ARG-ot FACC-edik hatványra emeli
AC3C–AF3E	44860–44862:	JMP $9033 EXP(FACC)
AF3F–AF41	44863–44865:	JMP $9409 COS(FACC)
AF42–AF44	44866–44868:	JMP $9410 SIN(FACC)
AF45–AF47	44869–44871:	JMP $9459 TAN(FACC)
AF48–AF4A	44872–44874:	JMP $94B3 ATN(FACC)
AF4B–AF4D	44875–44877:	JMP $8C47 FACC kerekítése
AF4E–AF50	44878–44880:	JMP $8C84 FACC abszolút értéke
AF51–AF53	44881–44883:	JMP $8C57 FACC előjel ellenőrzése
AF54–AF56	44884–44886:	JMP $8C87 FACC és MEM összehasonlítása
AF57–AF59	44887–44889:	JMP $8437 Lebegőpontos véletlen szám generálása
AF5A–AF5C	44890–44892:	JMP $8AB4 RAM MEM-et ARG-ba
AF5D–AF5F	44893–44895:	JMP $8A89 ROM MEM-et ARG-ba
AF60–AF62	44896–44898:	JMP $7A85 RAM MEM-et FACC-be
AF63–AF65	44899–44901:	JMP $8BD4 ROM MEM-et FACC-be
AF66–AF68	44902–44904:	JMP $8C00 FACC-t MEM-be

A BIT–LET karácsony szaktanácsadói asztalánál különösen a délelőtti órákban volt nagy a forgalom. Mint „Commodore-ügyeletes", néhány olyan kérdés mellett, amit lehetetlen megválaszolni („írtam otthon egy programot, nem megy, mit csináljak..." vagy „Elromlott a gépem, mi lehet a baja?"), számos érdekes problémával is találkoztam. A kérdések gyakorisága és fajsúlya alapján választottam ki azokat, amelyek esetleg nagyobb érdeklődésre tarthatnak számot.

Morvai László:

SZAKÉRTŐ VOLTAM

Különbözik-e egymástól a C 16, Plusz 4 és a C 64 gépi kódja? Már többször megírtuk (bár lehet, hogy az új géptulajdonosok csak most kezdik bújni a honi szaksajtót), hogy a két géptípus mikroprocesszora egymással teljesen kompatíbilis, vagyis a C 64-en megírt gépi kódú programot a C 16 és a Plusz 4 is megérti, sőt meg is próbálja végrehajtani, csakhogy a gépek egyéb szoftver és hardver különbségei miatt a futtatási kísérletek 99,9%-a teljes kudarccal végződik.

Hogyan lehet a C 16, Plusz 4 karakterkészletét módosítani? Azt, hogy a karakterkészlet a ROM-ban, vagy a RAM-ban helyezkedik el, a 65298 ($ff12) TED-regiszter 2. bitjének állapota határozza meg. Ha ez a bit magas (=1), a karakterkészlet a ROM-ban, ha alacsony, a RAM-ban helyezkedik el. A „kapcsolgatást" az alábbi BASIC utasításokkal végezhetjük el:

RAM : POKE 65298, PEEK(65298) AND 251
ROM : POKE 65298, PEEK(65298) OR 4

A karakterkészlet memóriabeli kezdőcímét a 65299($ff13) TED-regiszter 2–7. bitjei adják meg. A cím értéke alapállapotban 209 ($dl):

bit: 7. 6. 5. 4. 3. 2. 1. 0.
1 1 0 1 0 0 0 1
3216 8 4 2 1

Ez egy hatjegyű bináris szám

A 2–7. bitek által ábrázolt hatjegyű bináris számot 1024-gyel megszorozva a karaktergenerátor kezdőcímét kapjuk. Alapállapotban ez (32+16+4)∗1024, ami 53248 ($d000). Fentiek ismeretében a karaktergenerátor tetszés szerint áthelyezhető és módosítható.

Hogyan lehet megállapítani, hogy egy általunk ismeretlen, kazettáról beolvasott gépi kódú program hol helyezkedik el a memóriában? A 819 ($0333) címen kezdődik a szalagpuffer a C 16 és Plusz 4 gépeken. A puffer első két Byte-ja a kezdőcímet, a harmadik és negyedik Byte-ja a végcímet adja meg. Akik nincsenek barátságban a hexadecimális számokkal, a monitor használata nélkül is boldogulhatnak, az alábbi BASIC utasítások segítségével:

Kezdőcím: PRINT PEEK(820)∗256+PEEK(819)
Végcím: PRINT PEEK(822)∗256+PEEK(821)

C 64-en a szalagpuffer a 828 ($033c) címen kezdődik, a probléma hasonlóképpen oldható meg.

Melyik cím tartalma határozza meg C 16, Plusz 4 gépeken az előtér (szöveg) színét? Az 1339 ($053b) cím. A TED regisztereknek a grafika használata esetén van szerepük.

Hogyan lehet C 16, Plusz 4 gépeken a RUN/STOP és a RESET gombokat hatástalanítani? A STOP vektor a RAM $0326–$0327 (806–807) címein helyezkedik el és a $f265 címre mutat. Ha a vektort úgy módosítjuk, hogy a következő kódra mutasson (pl. $f266), a STOP gomb hatástalan lesz. Ezt megtehetjük a INC $0326 gépi kódú, vagy a POKE 806, 102 BASIC utasítással. Ha gépi kódú programunk van, átírhatjuk a STOP vektort úgy is, hogy a program startcímére mutasson, így elérhetjük, hogy a STOP lenyomására a program újra elinduljon.

A RESET hatástalanítása már nem ilyen egyszerű feladat hiszen a RESET vektor a ROM-ban, a $fffc–$fffd címeken helyezkedik el. Ha tehát egy programmal a RESET-et is hatástalanítani akarjuk, először ki kell kapcsolni a ROM-ot (STA $ff3f), majd a $fffc–$fffd címekre a RAM-ba beírni az új vektort, amely vagy egy RTS-re, vagy a program startcímére mutat. Mivel a ROM nem „él", a programunk által használt BASIC és KERNAL szubrutinokat az „akció" előtt át kell másolnunk a RAM területre. Megéri?

A fenti fogások programvédelemre akkor használhatók, ha a programunk autostartos, öninduló.

Szerkesztőségünktől sokan és gyakran kérdezik, hogy: – Tessék mondani, a C 16-os és Plusz 4-es gép

tényleg egyforma?

Egy darabig meggyőződéssel vallottuk, hogy lényegében igen, s ha az egyiken írott program nem fut a másikon, az csak programozói hanyagság eredménye lehet (lásd C-újság 3. szám – Hogy fut, hogy nem fut? című írás). Meglehetősen elbizonytalanodtunk azonban, amikor néhány héttel ezelőtt egyik külső munkatársunk azzal állított be hozzánk, hogy a két gép ROM listájában eltéréseket fedezett föl s meglobogtatta az általa kiprinteltetett listarészletet.

A különbség nem a két gép, a C 16, illetve Plusz 4 között van. Eddigi nyomozásunk eredményeként megállapítottuk, hogy mindkét géptípust kétféle KERNAL ROM-mal hozták forgalomba, tehát mindkét gépnél találtam mindkét típusú KERNAL-ból beszerelve.

ELSŐ KUNSZT

```
. CCCF   A5 79     LDA $79
. CCD1   D0 11     BNE $CCE4
. CCD3   A9 28     LDA #$28
. CCD5   85 79     STA $79
. CCD7   20 06 A9  JSR $A906
. CCDA   86 7A     STX $7A
. CCDC   84 7B     STY $7B
. CCDE   A0 28     LDY #$28
. CCE0   A9 0D     LDA #$0D

>CDBA FF FF FF FF FF FF FF FF :▲▲▲▲▲▲▲▲
>CDC2 FF FF FF FF FF FF FF FF :▲▲▲▲▲▲▲▲
>CDCA FF FF FF FF.FF FF FF FF :▲▲▲▲▲▲▲▲
```

```
. CCCF   A5 79     LDA $79
. CCD1   D0 11     BNE $CCE4
. CCD3   A9 28     LDA #$28
. CCD5   85 79     STA $79
. CCD7   20 06 A9  JSR $A906
. CCDA   86 7A     STX $7A
. CCDC   84 7B     STY $7B
. CCDE   A0 28     LDY #$28
. CCE0   20 BA CD  JSR $CDBA

. CDBA   A9 79     LDA #$79
. CDBC   91 7A     STA ($7A),Y
. CDBE   A9 00     LDA #$00
. CDC0   C8        INY
. CDC1   91 7A     STA ($7A),Y
. CDC3   60        RTS
```

Ez a rutin a DS$ előállítását és tárolását határozza meg. A különbség, hogy a DS$ végére az A esetben egy kocsi visszát, a B esetben pedig a DS$ pointerek címét teszi a rutin. Arra nem sikerült rájönnünk (eddig), hogy ez okozhat-e valamilyen problémát programok írásakor, futtatásakor.

MÁSODIK KUNSZT

```
. D8C1   AC 5D 05. LDY $055D
. D8C4   F0 0E     BEQ $D8D4
. D8C6   AC 5E 05  LDY $055E
. D8C9   B9 67 05  LDA $0567,Y
. D8CC   CE 5D 05  DEC $055D
. D8CF   EE 5E 05  INC $055E
. D8D2   58        CLI
. D8D3   60        RTS
. D8D4   AC 27 05  LDY $0527
. D8D7   A2 00     LDX #$00
. D8D9   BD 28 05  LDA $0528,X
```

```
. D8C1   AE 5D 05  LDX $055D
. D8C4   F0 0F     BEQ $D8D5
. D8C6   AC 5E 05  LDY $055E
. D8C9   B9 67 05  LDA $0567,Y
. D8CC   CE 5D 05  DEC $055P
. D8CF   EE 5E 05  INC $055E
. D8D2   58        CLI
. D8D?   18        CLC
. D8D4   60        RTS
. D8D5   AC 27 05  LDY $0527
. D8D8   EA        NOP
. D8D9   BD 28 05  LDA $0528,X
```

A billentyűzetpufferből hozza el a következő karaktert ez a rutin. A formai különbségeken túl lényeges eltérés a két módozatban az, hogy az A verzió nem használja az X regisztert, ha a funkcióbillentyű valamelyik karakterét hozza el, míg a B verzióban az X regiszterben, az elhozott karakter pozícióját találjuk a hozzátartozó karakterstringben. A B verzió ezenkívül törli a c bitet is, ha a funkcióbillentyűt használtuk. Mivel ezt a rutint használja a GETIN Kernal-rutin is, a funkcióbillentyű lenyomására adott válaszukban már eltérések vannak! Mivel a c bit magas állása hibaüzenetet jelent a kernal-rutinoknál, így a GETIN-rutint az A verzióban használva például hibaüzenetet kaphatunk, ha a funkcióbillentyűt lenyomjuk, és a hívás előtt véletlenül c-bitünk magas volt.
Sajnos BASIC-ben is jelentkezik a hiba az A verziójú gépeknél. Egy egyszerű sorral tudjuk bizonyítani állításunkat:

```
10 GETKEY A$: GETKEY B$
20 GOTO 20
```

Az A verziójú gépeknél a RUN után, ha egy funkcióbillentyűt megnyomunk, akkor azonnal leáll a program, összevissza kevert hibaüzenetet ír ki a 10-es sorban, míg a B verzióban a program simán eljut a funkcióbillentyű lenyomása után a 20 sorra, ahol RUN/STOP-pal meg tudjuk szakítani a normál módon. Az A verziójú gépeknél így biztonságosan a GETKEY funkciót nem használhatjuk, helyette érdemesebb a GET-tel dolgozni.

A példaprogram egyébként kiválóan alkalmas arra is, hogy ki-ki megállapítsa, melyik verziójú géppel rendelkezik.

HARMADIK KUNSZT

```
. EE21   48        PHA
. EE22   A5 AD     LDA $AD
. EE24   8D E8 05  STA $05E8
. EE27   A9 82     LDA #$82
. EE29   4C EC EC  JMP $ECEC
```

```
. EE21   48        PHA
. EE22   EA        NOP
. EE23   EA        NOP
. EE24   8D E8 05  STA $05E8
. EE27   A9 82     LDA #$82
. EE29   4C EC EC  JMP $ECEC
```

Ezt a TKSA-Kernal rutin tartalmazza. A két program különbségéből fakad, hogy az A verzióban, ha a TKSA-Kernal rutint meghívjuk, akkor a soros busz output-regiszterét kötelezően az $AD cím fogja beállítani, és akármi lehet a meghívás előtt az akkumulátor tartalma, míg a B verzióban a TKSA meghívásakor az akkumulátornak kell tartalmaznia az output-regiszter beállításához szükséges adatot, és az $AD cím pedig nem játszik szerepet a rutin használatakor. A kétféle rutin közötti különbség a KERNAL működésében nem okoz problémát, mivel ahányszor a TKSA rutint meghívja, előtte az akkumulátorba betölti az $AD cím tartalmát, így a BASIC mindig egyformán reagál. **Azonban, ha egy általános használatra készített gépi kódú programban használni kívánjuk a TKSA-rutint, akkor csakis a következőképpen használjuk, hogy mindegyik gépen működjön:**

```
LDA $AD
JSR $FF96
```

KÖVETKEZTETÉS

Mivel a különbségből fakadóan a legradikálisabb hibát az A verziójú gépeknél találtam, így véleményem szerint a B verzió egy javított kiadás lehet. Ez még nem jelenti azt, hogy a ROM-program más hibát nem tartalmaz, csak éppen nem derítettük fel. Például az RCLR függvény nem úgy adja vissza a színt, mint ahogy COLOR-ral beállítjuk, hanem a 16-os szín helyett 0-át ad vissza, a többi azonban megegyezik.

```
10 COLOR 1,16 : A=RCLR(1)
20 COLOR 1,1 : COLOR 2,A
```

Ha tehát ezt a programot lefuttatjuk, a 20. sor megáll ILLEGAL QUANTITY hibaüzenettel, amely egy logikátlan reakciót jelez a gép részéről. A ROM-program javított változatát is leközlöm, amely pontosan visszaadja a beállított színeket, és nem okozza a program megakadását a fenti esetben. (Ez a módosítás azonban csak elméleti, hiszen a ROM-ban nem tudjuk elhelyezni, csak a KERNAL kicserélésével.)

```
. BFB5   18        CLC
. BFB6   69 00     ADC #$00
. BFB8   29 0F     AND #$0F
. BFBA   A8        TAY
. BFBB   4C 81 9A  JMP $9A81
```

```
. BFB5   18        CLC
. BFB6   69 00     ADC #$00
. BFB8   29 0F     AND #$0F
. BFBA   A8        TAY
. BFBB   4C 81 9A  JMP $9A81
```

A fentiek mutatják, hogy a ROM programja még tartalmazhat észrevétlen hibákat. Ha olvasóink találtak ilyeneket, kérjük, közöljék, hogy közösen próbáljunk megoldásokat találni, s így legalább tudjuk, hogy miért viselkedik olykor érthetetlenül a gépünk.

Tóth Kornél

Meszlényi Zoltán

MAGÁN – BASIC

A C 16, Plusz 4 gépek által használt 3.5 BASIC-változat igazán nem szorul bővítésre, vagy kiegészítésre. Érdemes mégis ezzel foglalkozni. Ha BASIC-programban saját gépi kódú rutinokat szeretnénk használni, a leghatékonyabb és legelegánsabb eljárás a BASIC bővítése új kulcsszavakkal, ugyanis az új utasítások beírhatók a programba, végrehajthatók, listázhatók és ugyanúgy rövidíthetők, mint az alap BASIC kulcsszavai.

Az interpretert felkészítették arra, hogy a $FE tokent és az azt követő Byte-ot felhasználói tokenként értelmezze. Ennek az eljárásnak előnye, hogy sokkal áttekinthetőbb a programlista, gyorsabb a végrehajtása, mintha SYS utasítással vagy USR-függvénnyel aktivizálnánk a gépi kódú programrészeket. Az eljárást csak egyszer kell pontosan kidolgozni, utána tetszés szerint bővíthető.

Tekintsük át nagy vonalakban, hogyan kezeli a BASIC interpreter a programsorokat. Amikor a billentyűzeten begépelünk egy utasítást, az először egy bemeneti tárolóba, pufferbe kerül. A RETURN billentyű lenyomásával megkezdődik a feldolgozás. Ha az első karakter betű, az interpreter a puffer tartalmát parancsként értelmezi és ha lehetséges, végrehajtja. Ha pedig szám, akkor programsorként kezeli. A programba nem azok a karakterek kerülnek, amelyeket begépeltünk, hanem BASIC kulcsszavak helyett közbenső kódok, ún. tokenek, a többi karakter helyett ASCII kódok. A tokenizálást végző rutin a $8953 címen található. A program végrehajtása során a $8C25-ön kezdődő rutin a token alapján egy táblázatból ($8383) választja ki azt az ugrási címet, ahová a vezérlés kerül. A program kilistázásakor a gépnek ismét elő kell állítani a tokenekből a kulcsszavakat (LIST rutin a $8AFF címen). Mindhárom rutin tartalmaz egy-egy indirekt ugrást a felhasználói tokenek kezeléséhez. A $30C–$30D vektor a tokenizáláshoz, a $30E–$30F vektor a listázáshoz, a $310–$311 vektor a végrehajtáshoz használható. Bekapcsolás vagy RESET után mindhárom vektor a megfelelő JMP utasítást követő címet tartalmazza.

Lássuk példaként, hogyan lehet a C-újság 1. számában megjelent BASIC bővítő program felhasználásával új utasításokat definiálni. Gépeljük, vagy töltsük be a programot, majd futtassuk. Toljuk feljebb a BASIC kezdetét: POKE 44,17:NEW. Kapcsoljunk át monitorba, T 600 667 109D utasítással helyezzük a programot a BASIC területre. Az OLD funkciót megvalósító rutin mielőtt meghívná a BASIC CLEAR-t, törli a visszatérési címet. Eljárásunkban nem SYS utasítással hívjuk ezt a rutint, ezért szükségünk van a visszatérési címre. A következőképpen módosítsuk:

```
A 10FA NOP
A 10FB NOP
```

Az interpreter $818E címen kezdődő táblázata valamennyi BASIC kulcsszót tartalmazza oly módon, hogy a szavak utolsó karaktere SHIFT-elt karakterkódként szerepel ($80-nal magasabb a normál kódnál). A táblázat végét 0 Byte jelzi. Foglaljuk hasonló táblázatba új utasításainkat.

```
>1053 42 4C 4F 41 C4 42 53 41 : BLORDBSA
>105B 56 C5 53 57 41 D0 4F 4C : VESWAPOL
>1063 C4 00 00 00 00 00 00 00 : D.......
```

A vezérlésátadás saját rutinjainkra RTS utasítással történik, ezért – mint az interpreter megfelelő táblázatában – **következő táblázatunkban is a kezdő címeknél 1 Byte-tal alacsonyabb értékek találhatók.**

```
>1080 9C 10 A8 10 BD 10 EE 10 : ..<.=./.
```

írjuk meg ezt a programot, amely saját táblázatunk alapján azonosítja a tokeneket.

```
1025 LDA #$10
1027 LDY #$53        ; a táblázat kezdö cime A-ba és Y-ba
1029 JSR $8A07       ; token azonositó rutinra
102C BCC $1032       ; kulcsszó nem egyezik
102E INY             ; mutató növelése
102F CLC             ; USER token jelzése
1030 BCC $1035       ; feltétlen ugrás
1032 JSR $0479       ; CHRGOT
1035 JMP $896A       ; vissza a ciklusba
```

A $30C–$30D vektorba írjuk be ennek a rutinnak a kezdőcímét címformátumban: $25 $10 Térjünk vissza BASIC-be. Egy rövid programmal ki is próbálhatjuk az eddigieket.

10 A$="A":B$="B":SWAP(A$,B$):PRINTA$;B$

Ismét MONITOR és M1100-al láthatjuk az eredményt.

Lássunk hozzá a BASIC LIST-rutin kiegészítéséhez:

```
1038 TAX             ; token az X-be
1039 LDY #$10        ; kulcsszó táblázat
103B STY $23         ; kezdöcime ($1053)
103D LDY #$53        ; $22-$23-ba
103F STY $22
1041 JMP $8B9E       ; vissza a ciklusba
```

Most kell átírni a $30E–$30F vektort: $38 $10. Ha eddig nem hibáztunk, BASIC-ből kiadhatjuk a LIST parancsot.

A következő rutinunkra akkor adódik vezérlés, ha a végrehajtás során az interpreter $FE tokent talál:

```
1044 SBC #$80        ; 128 levonása
1046 ASL             ; cimkiszámitás
1047 TAY
1048 LDA $1081,Y     ; rutin cime -1
104B PHA
104C LDA $1080,Y     ; táblázatból
104F PHA             ; verembe
1050 JMP $0473       ; CHRGET és rutinhivás RTS-sel
```

Ennek működését is kipróbálhatjuk, ehhez a $310–$311 vektorba írjunk $44 $10-et.

Meg kell még írnunk azt a programot, mely lefoglal egy lapot a BASIC területből, és átírja a vektorokat.

```
1005 LDA #$11
1007 STA $2C         ; BASIC kezdet
1009 LDY #$05
100B LDA $101F,Y
100E STA $030C,Y
1011 DEY
1012 BPL $100B       ; vektorok beállitása
1014 JSR $8A7B       ; BASIC NEW
1017 LDA #$00
1019 STA $1100       ; 0 Byte a BASIC elejére
101C JMP $8BD3       ; interpreter ciklusra
>101F 25 10 38 10 44 10
```

Amennyiben sikeres volt eddigi ténykedésünk, összeállt a teljes program. Érdemes kimenteni, hiszen bármikor kiegészíthetjük újabb rutinokkal. Ehhez csupán a kulcsszó táblázatba kell beírni az új utasítást és a címtáblázatot kiegészíteni az új rutin kezdőcímével.

LEMEZ-MEGHAJTÓK

Sokszor, sokan kérdezik, hogy akkor végül is mi mivel működik? És melyik, mivel olvasható? Érthetőbben, melyik típusú Commodore gép melyik lemezmeghajtót tudja használni, és a felírt lemez más gépekkel, más meghajtóval olvasható-e? A hosszú ideig számunkra is zavaros képben megpróbálunk most egy kis rendet teremteni.

VC 1541

Ez az alaptípus, amit eredetileg a C 64-eshez fejlesztettek ki. A külföldi kereskedelemben elterjedt gyorsítók általában ehhez a meghajtóhoz kapcsolódnak. Több éves változatlan konstrukció után néhány hónapja megjelent a VC 1541C nevű modernebb gép, amely csak abban különbözik az alaptípustól, hogy az elektronikában több chipet egyesítettek, így kisebb panelen fér el ugyanaz, jobb a hűtése (C= újság 86/6).



MŰSZAKI ADATOK

Lemezhasználat:	egyoldalas
Csatlakozás:	soros csatlakozó
Átviteli sebesség:	300–400 Byte/sec
Teljes lemezkapacitás:	174 848 Byte
Adatkapacitás:	168 656 Byte
Tartalomjegyzék:	max. 144 file
Sávok száma a lemezen:	35
Szektorok száma sávonként:	17–21
Szektorok száma lemezenként:	683 664 szabad szektor
Byte-ok száma szektoronként:	256

ÁRAMKÖRI ELEMEK

Processzor:	6502
Input-output vezérlés:	2 db 6522
Memória:	4 db 2114, 2KByte RAM

MÉRET

Magasság:	97 mm
Szélesség:	200 mm
Hosszúság:	374 mm

KOMPATIBILITÁS

Közvetlenül használható a következő számítógépekkel:	C 16, C 116, VC 20, C 64, Plusz 4, C 128
Az ezzel készült lemez biztosan olvasható a következő más típusú meghajtókkal:	VC 1550, VC 1551 VC 1570 és VC 1571 (64-es üzemmódban)

MEGJEGYZÉS

Hallottunk a VC 1540-es típusszámú változatról ezzel azonban még soha sehol nem találkoztunk. A leírások szerint semmiben sem különbözik a 1541-estől. Ha egyáltalán létezik!

VC 1551

Az új gépcsaládhoz (C16–Plusz 4) kifejlesztett változat, amely külsőre csak színében és csatlakozójában tér el a 1541-től, a gép elektronikája azonban egészen más. A beépített mechanika nem változott. Működési elve: az adatok több vezetéken való egyidejű mozgatása (párhuzamos csatlakozás), így sebessége nagyobb elődjénél.



MŰSZAKI ADATOK

Lemezhasználat:	egyoldalas
Csatlakozás:	párhuzamos csatlakozó
Átviteli sebesség:	1600 Byte/sec
Teljes lemezkapacitás:	174 848 Byte
Adatkapacitás:	168 656 Byte
Tartalomjegyzék:	max. 144 file
Sávok száma a lemezen:	35
Szektorok száma sávonként:	17–21
Szektorok száma lemezenként:	683 664 szabad szektor
Byte-ok száma szektoronként:	256

ÁRAMKÖRI ELEMEK

Processzor:	6510T
Input-output vezérlés	6523A vagy 6525
Memória:	23128, 16 KByte ROM 4016, 2 Kbyte RAM

MÉRET

Magasság:	97 mm
Szélesség:	200 mm
Hosszúság:	374 mm

KOMPATIBILITÁS

Közvetlenül használható a következő számítógépekkel:	C 16, C 116, Plusz 4 VIGYÁZAT!!! MÁS GÉPEKKEL SEMMILYEN MÓDSZERREL SEM HASZNÁLHATÓ!!!
Az ezzel készült lemez biztosan olvasható a következő más típusú meghajtókkal:	VC 1550, VC 1541 VC 1570 és VC 1571 (64-es üzemmódban)

MEGJEGYZÉS

Hasonló módon az előzőekhez hallottunk már VC 1550-es típusról, találkoznunk azonban még nem sikerült vele!

VC 1571

A C 128-ashoz kifejlesztett alapvetően új lemezmeghajtó. A C 128-as gép konstrukciójához kapcsolódóan többféle üzemmódban képes működni. Egy szoftver megoldással azonosítja a hozzákapcsolt számítógép típusát és üzemmódját, és ennek megfelelően dolgozik. A három üzemmód a következő:

64-es üzemmód	ilyenkor pontosan úgy viselkedik, mint a VC 1541
128-as üzemmód	ilyenkor dupla sebességgel és kétoldalas üzemmódban dolgozik, a formátuma azonos a VC 1541-esével, de a lemez mindkét oldalát használja
CP/M üzemmód	alapvetően más alkalmazási mód, csak a C 128-assal használható, más nagyobb gépek által olvasható formátum, 500 Kbyte-os lemezkapacitás.

MŰSZAKI ADATOK

128-as üzemmódban (64-es üzemmódban azonosak a VC 1541-esével)

Lemezhasználat:	kétoldalas
Csatlakozás:	soros csatlakozó
Átviteli sebesség:	600–800 Byte/sec
Teljes lemezkapacitás:	2 × 174 848 Byte
Adatkapacitás:	2 × 168 656 Byte
Tartalomjegyzék:	max. 144 file
Sávok száma a lemezen:	2 × 35
Szektorok száma sávonként:	17–21
Szektorok száma lemezenként:	1366 1328 szabad szektor
Byte-ok száma szektoronként:	256

ÁRAMKÖRI ELEMEK

Processzor:	6502A
Input-output vezérlés	2 db 65C22A
Memória:	23256, 32 KByte ROM 4016, 2 Kbyte RAM

MÉRET

Magasság:	76 mm
Szélesség:	216 mm
Hosszúság:	346 mm

KOMPATIBILITÁS

Közvetlenül használható a következő számítógépekkel:	
64-es üzemmódban:	C 16, C 116, VC 20, C 64, Plusz 4
128-as üzemmódban:	C 128

A 64-es üzemmódban ezzel készült lemez biztosan olvasható a következő meghajtókkal: VC 1570, VC 1551, VC 1541

MEGJEGYZÉS

Hasonló módon az előzőekhez hallottunk már VC 1570-es típusról, de most tudjuk is, hogy ez mit jelent! Elektronikája mindenben megegyezik a 1571-essel, de a beépített meghajtó mechanikája csak egy oldalon kezeli a lemezeket. Rendelkezik mindhárom üzemmóddal, a 64-es és a 128-as üzemmód között azonban csak sebességben van eltérés.

BIRD MOTHER

Rovatunkban ezúttal ismét játékprogramokat futtatunk. Az 1986/5. számunkban már néhány szóban bemutatott C–64-es játékok közül választottunk. A BIRD MOTHER (Anyamadár) program hazai „mesterek" munkája, és ha minden igaz, jelenleg is kapható a boltokban. A programfutamban öt akadályt állítottunk föl:



Király Zoltán,
23 éves,
az ELTE
matematika szakos
hallgatója,
gyakorlott programozó.

ALAPÖTLET — 4

Ebben elsősorban az alapötlet újdonságát kívánjuk értékelni, ütőképességét, meglepő voltát, eredetiségét.

Kedves, békés játék, lövöldözés nélkül. A játékban elérendő 3 cél jópofának tűnik. Bár a konkrét játékötlet valószínűleg új, a játék elemei mégis elég nagy hasonlóságot mutatnak más játékokkal.

MEGVALÓSÍTÁS — 3/4

Ez az akadály éppen arról szól, amit a neve is jelez. Milyen az alapötlet részletekbe menő megvalósítása, a részötletek színvonala, azaz az elméletben megfogalmazott alapötlet kidolgozása.

Néhány hiba: amikor a játékos először kerülne fel a pontszám táblázatra, neve helyére RUN kerül fel.
Ha a madár kirepül a képből, néha igen nehéz visszahozni. Például: felfelé kirepülés esetén (ha véletlenül a játékos hosszabb ideig tartja a botkormányt a felfelé irányban), akkor lefelé nem is lehet visszacsalni a madáranyát. Ez csak újabb felfelé mozgatással sikerül. Ráadásul szerencsétlen állat a képernyőn kívül is meghalhat.
Néhány játék után úgy tűnik, hogy a fészeképítés után a játék véletlenszerűen adja a jutalompontokat

GRAFIKA — 4/5

Ezt nem kell külön magyarázni, a játékprogramok egyik leglényegesebb elemének minősítését értelemszerűen nem hagyhattuk el.

Sok szép megoldással találkozhatunk. Az ágak és kövek esése nagyon szép, a nagyobb kövek gyorsabban esnek, a 2. fordulóban a bot repülése is jól kidolgozott.

HANG — 4/5

Ez sem maradhat ki.

A szép, kellemes zenei repertoárból nekem csak a madár elhalálozásakor játszott „gyászhimnusz" nem tetszett.

KEZELHETŐSÉG — 3/4

Sem a játék készítői, sem futtatói számára nem lehet elhanyagolható szempont, hogy egy játékkal hogyan boldogul el az egyszerű, számítógéphez esetleg nem is értő játékos. Alapelvünk, hogy az a jó szoftver, amely önmagát magyarázza, s kezelése egy értelmes ember, vagy akár versenybíró számára semmiféle gondot nem jelent.

Néhány hibát már a megvalósításnál is leírtam. De bőven lehet folytatni a listát. Például azt, hogy a madár mikor kapja el a csőrével az ágat, vagy a legyet, igen nehéz kitapasztalni, s még azután se mindig könnyű irányítani. Ha jobbra-balra irányt váltunk, a madár úgy fordul meg helyben, hogy közben a csőre elég sokat elmozdul – ezt felfoghatnánk akaratlagos nehezítésnek is, de nem szívesen fogom föl annak.

ÁTLAG

GYERMEKFUTAM

Kell mondanunk? A gyerekek csodálatosak. Sajátos magabiztossággal ülnek le a számítógép elé, megérzik az apró képpontok minden rezdülését, a joysticket rángatva feloldódnak a képernyő-mesékben. Ebben is különböznek tőlünk. És ezért is fontos az ő véleményük, nemcsak azért, mert – mint legfontosabb játékszoftver-fogyasztók – ők kunyerálják ki zsebünkből a programra valót.
A játékidőt 90 percben maximáltuk. Mielőtt elmentek, kértük őket, mondják el egymondatos véleményüket.



Olcsvári Zoltán,
17 éves,
a Márvány utcai
Finommechanikai
és Műszeripari
Szakközépiskola
3-os tanulója.



Tóth István,
31 éves a BME Közművelődési
Titkárságának
propaganda-szerkesztője.
A számítógépet
munkája során
rendszeresen használja.

3 **1** **2,66**

Nem igazán eredeti, viszont sok hasonló játékkal összevetve ez még a jobbak közé tartozik.

4 **2** **3,16**

Igazán barátságos program, kellemes, és feszültség nélkül játszható. Persze érdekesebb lenne, ha minden fázis több változást mutatna. Az sem ártana, ha nem kellene mindig az első feladattól újrakezdeni, hanem az utolsó megoldott akadálytól folytatni lehetne. Fölöslegesnek tartom, hogy a madár kirepülhet a képernyőről. A második fordulóban stílusosabban is meg lehetett volna választani a madár ellenfelét. (A bot egy kicsit gyerekesnek tűnik.)

Az alapötletet és a további négy akadály leküzdését nem tartom figyelemreméltónak. A madáranyáról Wodehouse szavain túl nincs is több mondanivalóm: „aszaltszilva tetemesen feljavított zamattal; még mindig olyan íze van ugyan, mint a gépolajba áztatott csomagolópapírnak, de egy jobb minőségű csomagolópapírnak..."

4 **3** **3,83**

Szép, hangulatos grafika. Tetszik, hogy a természetet ábrázolja, és nem a fantasztikumot. Jó, hogy ez az „állati" program mégis csak emberi környezetben játszódik. Tetszik az is, hogy az állatok fennmaradásának valóságos gondjait jeleníti meg.

5 **2** **3,83**

Teljesen a játék hangulatával él, különösen a madár halálakor hallható szomorú dallamot tartom telitalálatnak. Egyszóval a hang – lehet, hogy csak azért, mert nem volt még hasonlóval dolgom – klasszis.

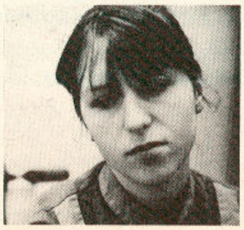
4 **2** **3,16**

Két dologgal volt bajom. Az egyik, hogy amikor megfordítom a madarat, a csőre helyzete megváltozik, a másik pedig, hogy az újszülött madarak túl picik.
A program leírását elégségesnek tartom, könnyen átlátható, és játszható a játék.

3,33



Tollas János (9 éves)
78 perc
Ha a nagyi nem jött volna értem, egész nap ezzel játszottam volna.



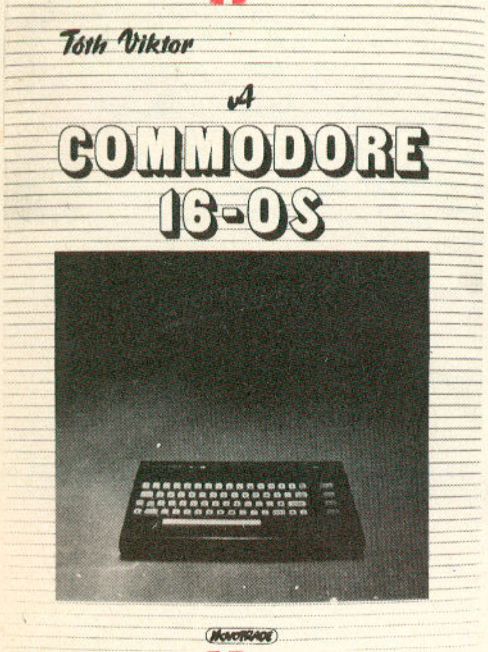
Borsodi Nikolett (13 éves)
38 perc
Lehet, hogy hülyeség, de sajnálom a madárfiókákat.

Tóth Viktor

A Commodore 16-os belső felépítése

Novotrade 1986. 99 Ft

Melyik volt előbb? A tyúk vagy a tojás? Miért írom ezt? Azért, mert a két könyv megjelenése között csupán néhány nap telt el. A témájuk hasonló – tulajdonképpen mindkettő gerincét a ROM-lista alkotja.



Ismerkedjünk meg közelebbről velük. A könnyebbség kedvéért Tóth Viktor könyve legyen TV, Erdős Iván könyve pedig EI. Először nézzük azt a részt, amely mindkét könyvben közös, a ROM-listát. TV a C 16-os listáját közli. EI pedig a Plusz 4-ét, különbség azonban ennek ellenére sincs a visszafejtett részben, mert EI kihagyta azt az „extra 32 KByte-ot", amelyik nincs meg a C 16-ban. Mindkettőben a BASIC és KERNAL ROM szerepel. **Sajnos, egyik listára sem lehet rámondani, hogy fekete vagy fehér, mindkettőben vannak hibák és hiányosságok. Nézzünk ezekre néhány példát.**

A TV-félében növekvő sorrendben vannak a memóriacímek, ezt azért hangsúlyozom, mert EI-ben ez nem jött össze egészen, ugyanis a könyvet rosszul fűzték össze (sajnos nemcsak egy példányt).

Mindjárt a ROM-lista elején TV-ben van egy visszafejtési „furcsaság", amilyenekkel a továbbiakban is találkozhatunk, ugyanis a 802A 71 98 71 94 sor egy „kezdő" számára magyarázat nélkül nehezen érthető. Az EI-féle változatban ez így néz ki:

802A 71 98
802C 71 94,

természetesen magyarázattal kiegészítve, ami szerintem jobban illeszkedik az összképbe.

Menjünk tovább. A 86CC-n lévő utasítás magyarázó szövege EI-ben „grafikus képernyő", TV-ben „video reset". A cím környezetéből világosan kiderül, hogy TV gondolta jól.

A magyarázó szövegben EI hol „jelző"-t, hogy „pointer"-t emleget, míg TV következetesen „kapcsoló"-t. Megjegyzem, EI írja is a bevezetőben, hogy az eredeti angol kifejezést használja, vagy ha van, akkor a magyar honosított változatot.

A TV-féle listában a „Műveleti hierarchia és címek (–1)"-nél a címek hibásak, mert nincs levonva az egy. Az EI-félében ez helyesen szerepel.

Mindkét változatban itt-ott hiányos a magyarázószöveg. Ha a kettőt összevetem, akkor sok esetben kiegészítik egymást.

Sorolhatnám még a hibákat, hiányosságokat, de nincs értelme. Sajnálom, hogy a két szerző nem együtt dolgozott, mert pontosabb, jobb ROM-listát készíthettek volna. Mégis, melyik a jobb, teheti fel jogosan a kérdést az olvasó. A TV-féle ROM-listára szavazok, annak ellenére, hogy mindkét listában körülbelül ugyanannyi hiba van, s teszem ezt azért, mert a TV-félében a lista áttekinthetőbb, tagoltabb, és a magyarázatok nyelvezete következetesebb.

Mindkét könyvben találhatunk még mást is. Az EI-félében van egy keresztreferencia lista, amely megmutatja, hogy egy adott című gépi kódú utasításra vagy adatra mely más utasítások hivatkoznak. Találhatunk még egy olyan részt, amely leírja, hogyan készültek a visszafejtések, milyen segédprogramok felhasználásával. Ezt a részt legfeljebb mint érdekességet lehet kezelni, hiszen a felhasználó úgysem fogja ezeket bepötyögni, hiszen a lista, ami ezekkel készíthető, benne van a könyvben.

Van még tartalomjegyzék, ami viszont hasznos, amit azért jegyzek meg, mert a másikban nincs.

A TV-féle könyvben található egy általános leírás a gépről; megismerkedhetünk a ROM működésével, a processzor felépítésével, utasításkészletével, az egyes utasítások végrehajtásának idejével. Olvashatunk a címzési módokról, a processzor interrupt és reset kezeléséről. A könyv tartalmazza a Kernal rutinok leírását, funkcióját, működését. Megtudhatunk néhány hasznos dolgot a rendszer hibakezeléséről. Bepillanthatunk a készülék video lehetőségeibe, a hangelőállítás műhelytitkaiba. Megismerkedhetünk a TED-del kívül-belül, amely az egyik legfontosabb alkatrésze a gépnek. Megtudhatunk valamit ki és beviteli lehetőségeiről, olvashatunk a BASIC-interpreter működéséről, megtalálhatók a monitor utasításai. Van kapcsolási rajz a gépről és RAM térkép is a könyvben.

Annak ellenére, hogy ennyi minden található egy kötetben, a leírások nem felületesek, a leírtak valóban hasznosíthatók. Ráadásul a könyv stílusa olyan, hogy olvasás közben szinte megelevenedik a gép belső világa, és értelmes lények gyanánt kommunikálnak egymással a gép egyes részei.

Balogh Jenő

Az LSI ATSZ Erdős Iván által írt kiadványának címe Commodore Plus/4, C–16, C–116 ROM lista – vagyis az, amiről a könyv szól.

Erdős Iván:

Commodore ROM-lista Plusz 4, C 16, C 116

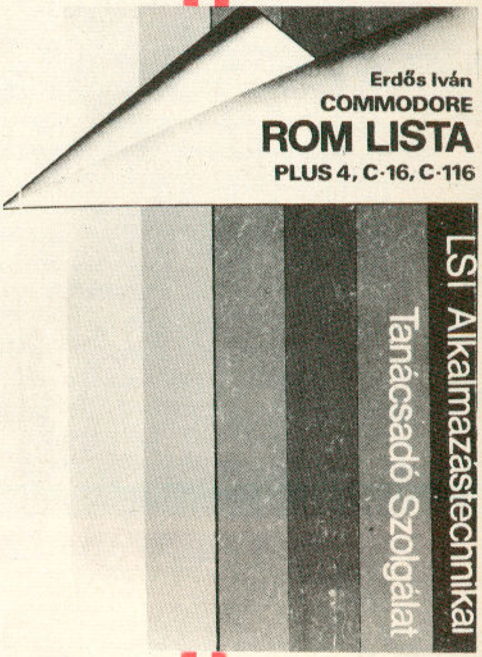
LSI 1986. 248 Ft

A két kötet szerkesztési koncepciója eltérő, mint ahogy az már a címadásból is kiderül.

A Novotrade RT szerzője, Tóth Viktor A Commodore 16-os belső felépítése címet adta kötetének, nyilván igazodva a más géptípusokhoz írt, hasonló témájú DATA BECKER kiadványokhoz. Felépítésében is igyekszik hasonlítani e könyvekhez, így átveszi azok meglehetősen logikátlan szerkezetét. Emiatt a kötet érdemi része a címnek megfelelően valóban a gép általános ismertetésével, belső szerkezetével foglalkozik, de ez alig teszi ki a teljes oldalszám egynegyedét – a főrészt jelentő ROM-lista pedig a függelékben kapott helyet.
Nyilvánvaló, hogy egy ROM-listát nem lehet tartalomjegyzékkel ellátni. Aki azonban leemeli a könyvesbolt polcáról Tóth Viktor kötetét, és felüti a tartalomjegyzéket, az érdekes címszavakat talál a géppel kapcsolatos tudnivalókról.
S ha nem tűnik fel neki, hogy a ROM-lista a 95. oldaltól a 426-ig terjed, azt hiheti, hogy egy alapszintű, a C–16-ot bemutató könyvet tart a kezében. Mondható, persze, hogy aki pusztán a tartalomjegyzék felületes átnézése alapján vásárol szakkönyvet, az megérdemli, hogy ne azt kapja, amit akar – a könyv szerkesztési elve azonban ezzel együtt is megkérdőjelezhető.
De ne akadjunk fenn ilyen formai gondokon, hanem térjünk rá a kötetek érdemi részére. Erdős Iván a bevezetőben mentegetőzik az esetleges magyartalanságok miatt, mondván, hogy a minél nagyobb információtartalom, a minél jobb érthetőség kedvéért a C–64 assembly programozásában meghonosodott kifejezéseket használják, még ha van is azoknak néha magyar megfelelője. Nos, összehasonlításul álljon itt mindkét lista commentjeiből (magyarázó szövegeiből) egy rövid részlet! A kezdőcím CE00, a megszakításkezelés belépési pontja.

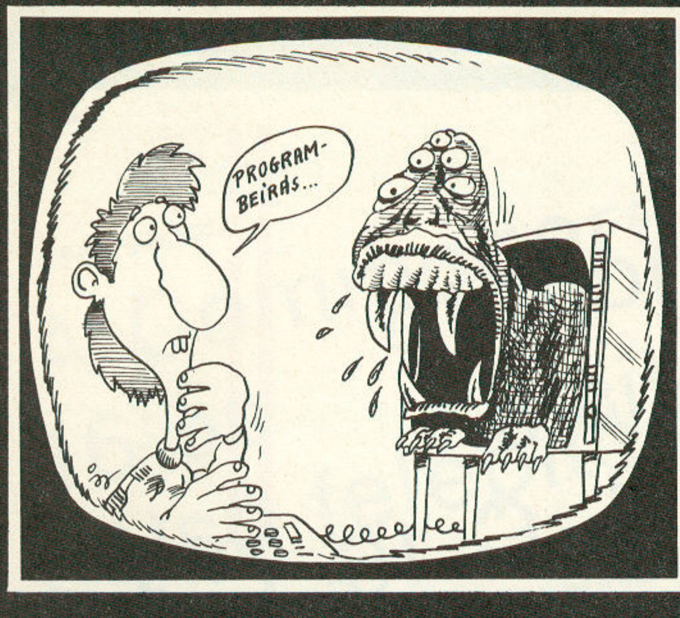
Erdős Iván	Tóth Viktor
	Veremmutató
	Előző státus
hardware megszakítás volt?	BRK utasítás?
nem	igen
alapfelt.: ceOe	IRQ vektor ($CE0E)
(ceOe)	BRK vektor ($F44C)
megszakítás-jelzők	TED interrupt regiszter
raszter?	Raszter?
nem	nem
RS-232 működik?	RS-232 (?)
nem	
RS-232 állapotjelző	

Talán ennyiből is kiderül, hogy melyik lista érthetőbb, magyarosabb, Erdős Iván kötete egyébként is jóval részletesebb magyarázatokat tartalmaz. Hogyan lehetséges ez annak ellenére, hogy az ő ROM-listája több mint száz oldallal rövidebb a másiknál? A magyarázat: ha nemcsak az oldalszámot, hanem azt is figyelembe vesszük, hogy melyik cég hány betűt (számot) nyomtatott egy oldalra, akkor kiderül, hogy az LSI listája közel kétszer olyan terjedelmű, mint a Novotrade-é. Így kiderül, hogy Tóth Viktor a selyempapírként áttetsző lapokon miért nem tudott ugyanannyi információt közölni.
Mi a helyzet a könyveknek a ROM-listát kiegészítő további részeivel? A Novotrade a kötet első, közel száz oldalán a gép általános jellemzőit, a processzorutasításokat, a KERNAL-rutinokat, a video- és hang-lehetőségeket, az I/O jellemzőket és a gépek Basic-jét ismerteti. Hasznos információk, jól megírva – csakhogy aki megveszi a kötetet, az a ROM-listáért teszi, így ezek az ismeretek nem igazán illenek ide. Erdős Iván könyvének tartalma szervesebb egységet képez: a főrészt – a listát – a „Műhelytitkok dióhéjban" c. függelék követi, ahol a szerző leírja, hogy hogyan is sikerült a ROM lefordítása, sőt részletesen be is mutatja a felhasznált fordítóprogramot. A kétfordulós fordítás első menetének, a hivatkozások felderítésének eredményét pedig külön is közli a kötet végén egy olyan táblázatban, amely minden felhasználó számára érdekes csemege lehet – ez a keresztreferencia-táblázat, amely az egyes címek egymásra való hivatkozását tartalmazza.



Aki tehát a C–16-os gépcsalád ROM-listájára kíváncsi, annak az LSI-kiadvány tanulmányozását javaslom – még ha jóval drágább is a Novotrade-énél –, akit viszont a géppel kapcsolatos általános tudnivalók érdekelnek, az vegye meg Tóth Viktor kötetét, és tekintse úgy, hogy kapott hozzá egy vaskos ROM-lista – mellékletet, amit szorgalmi feladatként sok-sok magyarázattal egészíthet ki.

Tallér József

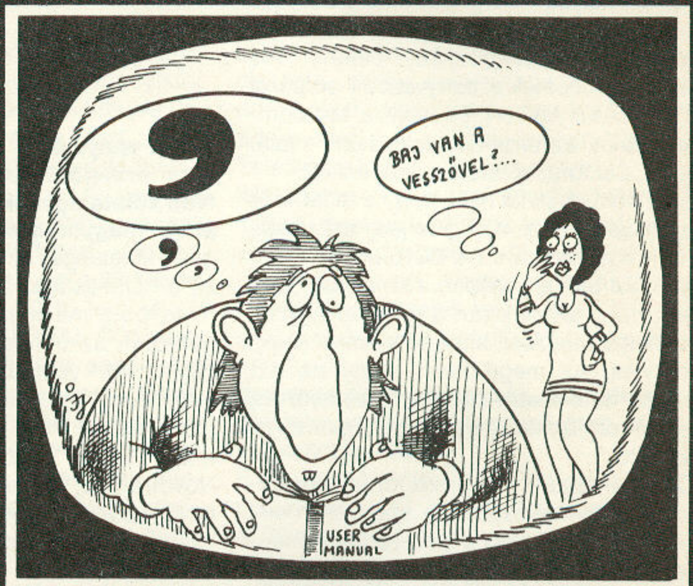


Mészáros Ferenc:

KÉPESREGÉNY

Egy óriási fehér doboz, benne új családtagunk, a C 16-os számítógép. Kellő áhítattal kivesszük a dobozból, beletesszük a táskába. És ezt a szép dobozt itt hagyjuk, Apa?! Dehogy hagyjuk, megkérjük a nénit, hogy kösse át és azt Te fogod hozni, így léptünk ki a bolt ajtaján. Büszkén vittük haza! Feleségem az ajtóban állva várta felbukkanásunkat. Diána ment elöl, feleségem rémülten kapott a doboz után: vigyázz, miért te hozod ezt? Már az üres dobozt is féltettük.
Nagyobbik szobánkban az asztal már üres felülettel várta, hogy kipakoljuk a C 16-ost. Az én kezemben addig egyáltalán nem volt személyi számítógép, nem ismertem, hogy milyen a belső felépítése, s habkönnyűnek találtam. Adaptere annál súlyosabbnak tűnt, termetéhez képest. Miközben pakoltam, Diána megsimogatta a gépet: jól van, te kis Komodorka!

Joggal kérdezheti az olvasó, hogy akkor hogyan képzeltük ezt az egészet?
Egy dolgot tudnunk kell, egy olyan család, ahol nincs iskoláskorú gyermek, legfeljebb a televízióban lát számítógépet, így mi mindent csak azután kezdtünk megtanulni, amikor már megvolt a gépünk is. Természetesen kinyitottuk a Felhasználói kézikönyv magyar nyelvű változatát és összeállítottuk a gépet. A bekapcsolási sorrendre – bárki bármit mond is – ügyelni kell és én ebből soha nem engedtem, feleségem ma is tőlem kérdezi, ha tanulni akar, hogy is kell, én már sorolom is: TV, adapter, gép. Kikapcsolni magától értetődően a gép, adapter, TV sorrend van.
Hát elkezdtük kóstolgatni a gépet. Persze mohó voltam, szerettem volna mindjárt rajzolni: karakterekkel kutyát-macskát, ebbe aztán hamar beleuntam. Elővettem a gép mellé kapott két ún. demo-kazettát. Betöltöttem az első programot, amely kiírja a képernyőre a C 16 színskáláját és aláfestésként lejátszik egy rövid zeneművet. Ez utóbbit, a zenei aláfestést csak hónapokkal később hallottam először, azon egyszerű okból, hogy nem adtam hangot a televíziónak. Ekkor találkoztam először a programbeírás rémével, amit úgy neveznek: nyomdai hiba. A könyv hetedik fejezetében van „A nagy Commodore 16 zenegép" című program, amit lelkesen bepötyögtettem. Mint kezdő számítógépes, igen sok „syntax errort" sikerült előállítanom, de ezekkel előbb-utóbb sikerült elbánni. Hanem az 1070. számú sorral nem bírtam. Pedig, amikor kilistáztam ezt a sort, azt láttam, hogy úgy van beütve, ahogy a könyv írja. Persze, én ekkor kizárólag felhasználóként funkcionáltam, a gép programnyelvéből még semmit sem értettem, másolni meg mindenki tud, aki olvasni tud! Hát akkor mit lehet itt tenni? Hopp, ez egy fordítás, van egy eredeti is. Elővettem az „User Manual" című angol gépkönyvet, összehasonlítottam a programlistákat. Akkor értettem meg, hogy amíg a magyar nyelvben nem súlyos hiba, ha egy koszos vessző hiányzik, egy BASIC program ezt nem türi el! Beütöttem a „megtalált" vesszőcskét a helyére, és a program elkezdett működni, őszintén szólva akkor ez nálunk szenzációnak számított: egy személyi számítógép, amelyik tud „zongorázni". Karácsony első napján jött hozzánk feleségem barátnője. Kislányom, Diána közölte: „Apa, mutasd meg keresztmamának a zongorát!" Keresztmama nézett a Bösendorferre, de Diána már mondta is: nem az, a Komodoré! A C 16 családtaggá vált, már büszkék voltunk rá.





Alulírott megrendelem a következő programokat a Pötyögő Szolgálattól:

PROGRAM SORSZÁMA		ÁRA
1.		,– Ft
2.		,– Ft
3.		,– Ft
4.		,– Ft
5.		,– Ft
6.		,– Ft
7.		,– Ft
8.		,– Ft
9.		,– Ft
10.		,– Ft

Összesen: db ,– Ft

A programokat saját kazettámra kérem.

A programokat saját lemezemre kérem.
(a megfelelő kockát x-elje be!)

A programokat a Szolgálat által biztosított Parrot SS, SD lemezre kérem (136,– Ft/db)

15' kazettára kérem. (26,– Ft/db)

POSTA költség (25,– Ft) ,– Ft

Összesen: ,– Ft

A fenti összeget befizettem az Országos Commodore Egyesület számlájára:
OPT XIII. Visegrádi u. 7/b.
MNB 217-98292
OTP 565-3610

NÉV:

EGYESÜLETI TAGSÁGI SZÁM:

DÁTUM:

....
aláírás

APRÓCSKÁK

Kérem, hogy a **Commodore Újság** legközelebbi számában jelenjen meg a következő szövegű apróhirdetés:
(minden oszlopba egy betűt írjon!)

Név: ....

Cím: ....

Egyesületi tagsági szám: ....

Feladandó az újság címére: **Commodore Újság**
Pozsonyi út 50. fsz. 4.
1133

A LEGYÜNK TÖBBEN akció részleteiről lapunk 3. oldalán olvashatnak. A lényeg, hogy aki igazolhatóan szervezője egy új szuperpáholy tagnak, vagy öt pluszpáholy tagnak, az fél évig tagdíjmentességet élvez egyesületünkben. A szervezés igazolása az alábbi szelvénnyel történik:

Az új belépő neve: ....

Az új belépő címe: ....

A tagsági díj befizetésének dátuma: ....
Melyik páholyba lépett – SZUPER – PLUSZ
(megfelelőt kérjük bekarikázni)
Alulírott igazolom, hogy tagságunk létrejötte

.... szervező munkájának köszönhető.

Az ő tagsági száma: ....
Kérem a LEGYÜNK TÖBBEN AKCIÓ-ban fenti nyilatkozatomat vegyék figyelembe.

....
SZUPER – PLUSZ – PÁHOLY új tagja
(illetve jogi személy esetén a cég képviselője)

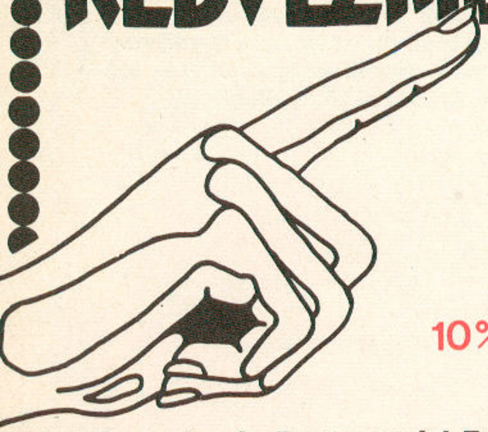
Természetesen az igazolás csak attól kezdve érvényes, amikor a nyilatkozó tagdíjának befizetéséről szóló utalvány is megérkezik Egyesületünkhöz.

PÖTYÖGŐ SZOLGÁLAT

Listánkban értelemszerűen a C16-tal jelölt programok Plusz 4-esen is futnak, a Plusz 4-gyel jelölt programok viszont igénylik a 64 KByte szabad memóriaterületet. A C64-S jelölés SIMON'S BASIC-ban írt programokat jelent.

SORSZÁM	GÉPTÍPUS	ELNEVEZÉS	ÁRA
Ø05	C16	Rajzoló program	3Ø.-
Ø06	C64	Átsorszámozo program	1Ø.-
Ø07	C64	HELP-TRACE	1Ø.-
Ø08	C16	Hangmemória	4Ø.-
Ø09	C64	Teke - Homecomputer	2Ø.-
Ø1Ø	C64	Csak kezdőknek!	1Ø.-
Ø11	C64	Billentyűkódok	1Ø.-
Ø12	C64	Autonumber	2Ø.-
Ø13	C16	BASIC bővítő	2Ø.-
Ø14	C64	BASIC billentyűzet	2Ø.-
Ø15	C64	Oszlopdiagram 1.	3Ø.-
Ø16	C64	Perifériavizsgálat	1Ø.-
Ø17	C64	Sprite mozgatás IRQ	2Ø.-
Ø18	C64	NOSCROLL	2Ø.-
Ø19	C64	Lemezmenű	1Ø.-
Ø2Ø	C64	Ékezetes nagybetűk 1.	1Ø.-
Ø21	C64	Ékezetes nagybetűk 2.	1Ø.-
Ø22	C64	Notesz	4Ø.-
Ø23	C16	TURBOBASIC	5Ø.-
Ø24	VC2Ø	Péter es Pál	3Ø.-
Ø25	C64	Mozart	3Ø.-
Ø26	C64	Segítség listázáshoz	2Ø.-
Ø27	C64	MERGE	2Ø.-
Ø28	C16	Billentyűzet kódoló	1Ø.-
Ø29	C16	Kódoló szabadon	1Ø.-
Ø3Ø	C64	AUTO-INSERT	1Ø.-
Ø31	C64	Kalandjáték	4Ø.-
Ø32	VC2Ø	TG Monitor	5Ø.-
Ø33	C16	Ablakok	2Ø.-
Ø34	C16	Karácsony	2Ø.-
Ø35	C64	Kockás zene	2Ø.-
Ø36	C64	Képújság	2Ø.-
Ø37	C64	Sprite editor	3Ø.-
Ø38	C16	SAVE-LOAD szignál	1Ø.-
Ø39	C64	Rubik kocka síkban	3Ø.-
Ø4Ø	C64-S	Oszlopdiagram 3.	2Ø.-
Ø41	VC2Ø	Winettou 1.	2Ø.-
Ø42	VC2Ø	Winettou 2.	2Ø.-
Ø43	C64	Stringrendező	2Ø.-
Ø44	C64	Lemezátnevező	1Ø.-
Ø45	C64	Óra	2Ø.-
Ø46	C64-S	Hiperkocka	2Ø.-
Ø47	C16	ASSEMBLER	5Ø.-

KEDVEZMÉNYEK



10% 10% 10%

A Szerszámgép Programozási Egyesülés felajánlja az Országos Commodore Egyesületnek, hogy Commodore 64-re készült szoftvertermékeinek árából az egyesület tagjai részére 10% kedvezményt nyújt az alábbiak szerint:

Program	Régi ár:	Új ár:
GEOPRO (geometriai számító- és rajzoló program)	7 000,–	6 300,–
ESZTER (optimális esztergálási paraméterek számítása	7 000,–	6 300,–
CSIGAPROG (csigahajtásmérető program)	5 000,–	4 500,–
Mindhárom program együttes vásárlása esetén	15 000,–	13 500,–

Érvényes: 1986. december 1–1987. február 28.
Igazolás: ennek a szelvénynek az elküldésével.

Sajnos az elmúlt félévben igen sok gondunk volt a vásárlási utalványokkal. Mindig reménykedtünk abban, hogy a következő lapszám már időben jelenik meg és mindig tévedtünk. Így a jogosan sértődött telefonok nagy mennyiségével kellett hónapról hónapra megküzdenünk. Minthogy idén már reálisabban mérjük fel a megjelenés pontosságának lehetőségét, ámde a telefonokat és Önök bosszantását szeretnénk elkerülni – azt a megoldást választottuk, hogy a tikettek jócskán előre lesznek dátumozva. Kérjük, senki se ijedjen meg, egyetlen hónap sem maradt ki, csak éppen mostantól hosszabb ideig lehet beváltani valamennyi utalványt.

50,– Ft-os vásárlási utalvány

50 Ft

Beváltható készpénzes vásárlás esetén a következő boltban:

2C áruház, XIII. Balzac u. 35.

Érvényes: 1987. március 31-ig.

NOVOTRADE

15%

Az ECONOMIX Közgazdász Egyetemi Kisszövetkezet 15% árkedvezményt ad a Sakkozó elefánt elnevezésű sakkoktató- és játékprogramjából a Commodore Egyesület tagjai részére.

Sakkozó elefánt	1290,– Ft
engedménnyel	1096,– Ft

Megrendelés: a szövetkezet címén:
1053 Budapest, Veres Pálné u. 36.
Telefon: 174-188
Érvényes: 1986. december 1–1987. február 28.
Igazolás: ennek a szelvénynek az elküldésével.

A Newline számítástechnikai vállalkozás 10% kedvezményt ad az egyesület tagjainak:

C 16 beépíthető 64 KByte memóriabővítő	1990,– Ft
16–64-es átkapcsoló	150,– Ft
beépítés munkadíja	490,– Ft
ROMTURBO 16	770,– Ft
együttes megrendelése esetén	3400,– Ft
árengedménnyel:	3060,– Ft

Jogosultak: a Plusz- és a Szuperpáholy tagjai
Igazolás: ennek a tikettnek postai elküldésével

Cím: Newline, 2220 Vecsés, Diófa u. 15.

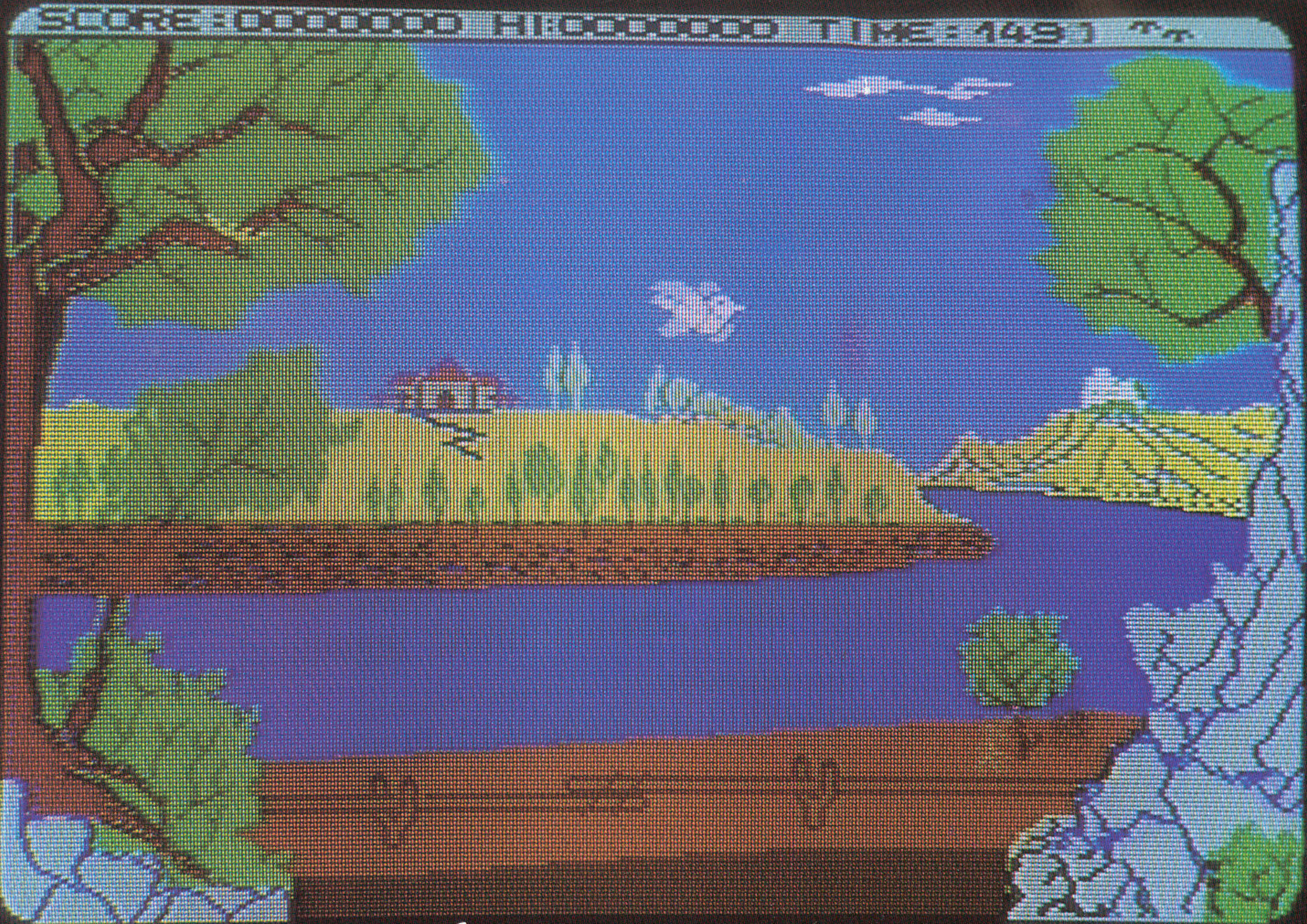
NEWLINE HARDWARE SOFTWARE

10%

A Novotrade-Fotoelektronik GT. az alábbi felsorolt szervizeiben mindenféle szervizszolgáltatás munkadíjából 10% kedvezményt ad egyesületi tagjainknak.
Jogosultak: valamennyi egyesületi tag
Határidő: nincs
A kedvezményt nyújtó szervizek:
Budapest V., Magyar u. 12–14. Telefon: 173-551
Pécs, Kolozsvár u. 20. Telefon: (72) 11-812
Szombathely, Szalonok u. 31. Telefon: (94) 14-519
Szeged, Székelysor 13.
Békéscsaba, Bartók B. u. 37.
Miskolc, Vologda u. 4. Telefon: (46) 17-011
Igazolás: a javítandó berendezés leadásakor egyesületi igazolvánnyal

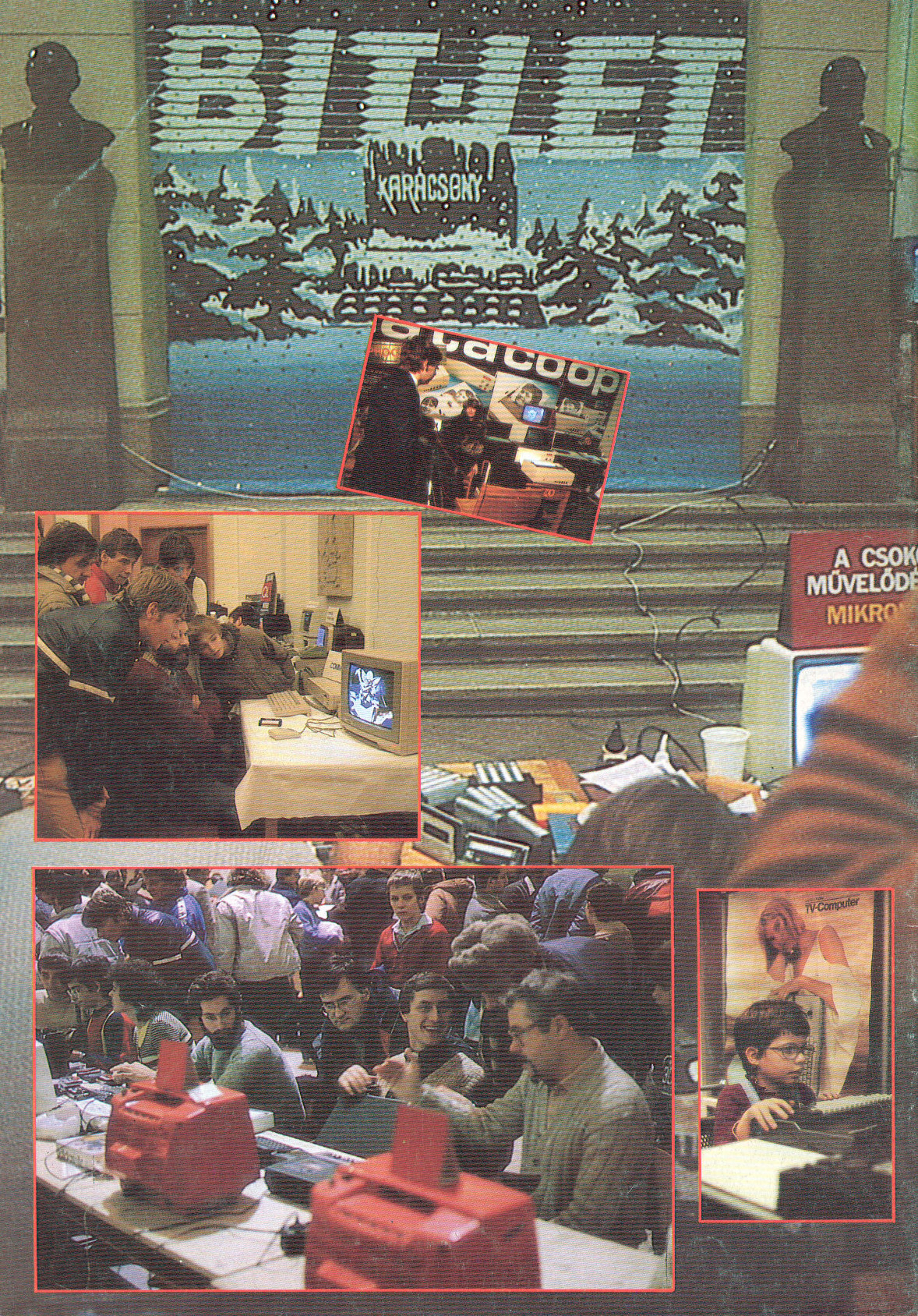
A kedvezmény többször is igénybe vehető.

FOTOELEKTRONIK GT NOVOTRADE



BIRD MOTHER

Programteszt a 28. oldalon


BITBIT
KARÁCSONY
TV-Computer